Mr. Andrijana Tomovska

# INFORMATIKA

PËR KLASËN VII PËR ARSIMIN FILLOR NËNTËVJEÇAR

**Autori:** Mr. Andrijana Tomovska

**Botues:** Ministria e Arsimit dhe Shkencës e Republikës së Maqedonisë së Veriut

#### Recensentë:

Prof. Dr. Marika Apostolova Trpkovska Zhaklina Naumovska Senar Fazlija

**Ilustrator:** Elena Trpçevska

**Lektor i botimit në maqedonisht:** Sllagjana Ginovska

**Përkthyes:** Prof. Dr. Nderim Zeqiri

Redaktor profesional: Prof. Dr. Nderim Zeqiri

**Lektor në gjuhën shqipe:** Prof. Fexhide Zeqiri

Me vendim për miratim të librit për lëndën Informatika për klasën VII në arsimin fillor nëntëvjeçar nr.26- 521/1 më 16.03.2020 e miratuar nga Komisioni Nacional për libra.

CIP - Каталогизација во публикација

Национална и универзитетска библиотека "Св. Климент Охридски", Скопје

## PARATHËNIE

Në përputhje me programin mësimor të krijuar dhe përcaktuar nga Byroja për zhvillimin e arsimit dhe e aprovuar nga Ministria e Arsimit dhe Shkencës e Republikës së Maqedonisë së Veriut, është krijuar ky libër shkollor për Informatikë për klasën VII për arsimin fillor nëntëvjeçar i detyrueshëm. Sipas programit mësimor zyrtar libri ka pesë tema.

Në temën e parë "Programi për llogaritje tabelare" nxënësit do të njihen me rrethinën e punës të Microsoft Office Excel 2016 dhe do të aftësohen për krijimin dhe rregullimin e tabelave, të kryejnë llogaritje matematikore me zbatimin e formulave dhe funksioneve dhe në fund paraqitjen grafike të të dhënave. Për ushtrime praktike të këtyre funksioneve të programit për llogarijte tabelare, janë krijuar pyetje përkatëse dhe detyra praktike.

Tema tjetër që do të trajtohet është "Njohja me konceptet informatike përmes zgjidhjes së detyrave konkurruese", ku nxënësit me ndihmën e detyrave konkrete do ta zhvillojnë të menduarit logjik, me qëllim që të gjejnë zgjidhjen e përshtatshme. Në zgjidhjen e detyrave konkurruese, nxënësit do të mësojnë lidhjen në mes të menduarit logjik dhe zgjidhjes me konceptet informatike dhe vendin e tyre gjatë punës me kompjuterë.

Në temën e tretë "Programimi i avancuar në një mjedis vizual" do të bëhet lidhje me temën e mëparshme, ku me ndihmën e logjikës do të krijohen blloqe të deklarimeve që do të kryejnë një aktivitet të caktuar. Në këtë mënyrë, duke zbatuar mjedisin vizual për programimin Scratch do të krijojmë programe të thjeshta, lojëra interaktive, tregime dhe të ngjashme.

Tema e katërt i kushtohet "Programimit përmes gjuhës programuese të strukturuar standarde". Për këtë qëllim do të përdorim mjedisin e integruar për programim Code::Blocks dhe gjuhën programuese C++. Përmes shembujve dhe detyrave praktike, nxënësit do të fitojnë aftësi në krijimin e programeve.

Në temën e fundit: "Të jetuarit onlajn" nxënësit do të aftësohen për krijimin e ueb-ditarëve, përkatësisht blogjeve me theks të veçantë në respektimin e rregullave për përdorim etik dhe sigurisë së Internetit.

Në këtë libër shkollor, punimet praktike janë baza për përvetësimin dhe zotërimin e njësive të reja mësimore dhe pikërisht për këtë është përgatitur pjesë e veçantë me pyetje dhe detyra të cilat do t'i mundësojnë nxënësit të zgjeroj njohuritë dhe shkathtësitë për temat e theksuara. Çdo njësi mësimore fillon me theksimin e koncepteve të reja kryesore me të cilat nxënësit do të hasin gjatë mësimit, dhe përfundon me rikujtime si rubrikë që tregojn për elementet më të rëndësishme të njësisë mësimore. Pas çdo mësimi janë dhënë pyetje të cilat shërbejnë për të përsëritur atë që është mësuar. Përmes mësimeve nxënësit do të kenë rikujtime, këshilla, dhe vërejtje të cilat janë vendosur në korniza të veçanta dhe janë posaçërisht të paraqitur në mënyrë grafike. Në vazhdim është dhënë lista me kuptimin e elementeve grafike që janë përdorur në libër.



Në fund të çdo teme është krijuar rubrika "Të përsërisim! Të ushtrojmë" si shtojcë e veçantë me pyetje dhe detyra për temat.

Në fund të librit është krijuar një fjalor i vogël me shpjegime të koncepteve profesionale dhe shtojcave që nevojiten gjatë përvetësimit të njësive mësimore. Në shtojcë të librit shkollor është krijuar një CD me të gjitha punët praktike të përcaktuara sipas folderëve të temave.

Ushtrimet, detyrat, punët praktike dhe pyetjet janë përgatitur në përputhje me moshën e nxënësve për klasën VII në arsimin fillor nëntëvjeçar dhe në përputhje me njohuritë e tyre paraprake të fituara në vitet e mëparshme shkollore në kuadër të kësaj lënde.

Autori

## PËRMBAJTJA

1. Hyrje në programin për llogaritje tabelare	6
1.1. Programi për llogaritje tabelare – Microsoft Office Excel 2016	7
1.1.1. Startimi i programit. Elementet në dritaren e punës	7
1.1.2. Puna me dokumente në Excel	10
1.1.3. Puna me elementet fillestare të listës së punëve	11
1.2. Puna me të dhënat në tabelë	14
1.2.1. Tipet e të dhënave	15
1.2.2. Formati i të dhënave numerike	16
1.3. Rregullimi i tabelës	19
1.3.1. Ndryshimi dhe rregullimi i të dhënave të futura	19
1.3.2. Aktivitete me kolonat dhe rreshtat	20
1.3.3. Ndryshimi i dimensioneve të kolonave dhe rreshtave	21
1.4. Formatimi i tabelës	
1.5. Plotësimi automatik i të dhënave në tabelë	23
1.6. Formulat dhe funksionet në programin për llogaritje tabelare	
1.6.1. Formula	31
1.6.2. Funksione	
1.7. Sortimi i të dhënave	
1.8. Krijimi i grafikut	40
TË PËRSËRISIM! TË USHTROJMË!	
2. Njohja me konceptet informatike duke zgjidhur detyra logjike	
konkurruese	48
2.1. Analiza dhe zgjidhja e detyrave logjike konkurruese	51
2.2. Lidhja e detyrës me konceptet e shkencës kompjuterike – koncep	otet
informatike	
2.2.1. Strukturat e te dhenave	55
2.2.2. Numrat binarë	57
2.2.3. Kriptografia	59
TE PERSERISIM! TE USHTROJME!	62
	65
3. Hyrje ne programim ne mjedisin vizual	65
3.1. Programimi grafik. Njonja me programin Scratch	
712 Nichia zao prietat there alere të Caratah	
5.1.2. Njonja me mjetet themelore të Scratch	68
3.1.3. Krijimi i programeve te tnjeshta ne Scratch	
5.2. Programet interaktive me ngjarje	
3.3. Shkruarja e programeve me situata problemore më komplekse	81
	87

4. Hyrje në programim përmes gjuhës programuese standarde të	20
41. Procesi i përpunimit të një programi	
4.2. Niohia me elementet themelore të mjedisit të integruar për	
programim	
4.2.1. Instalimi i Code: Blocks	
4.2.2. Mjedisi i punës të Code: Blocks	
4.2.3. Krijimi i një fajlli (skedari) të ri për kodin burimor	100
4.3. Paraqitja e kodeve të programore - të shembullit të gatshëm	102
4.4. Ekzekutimi i programeve-të shembullit të gatshëm	105
4.5. Elementet themelore të qjuhës programuese C++	
4.6. Deklarimet	112
4.6.1. Deklarimi për paragitje në ekran	
4.6.2. Deklarimi për ndarie të vlerave	
4.7. Përpunimi i programeve	116
4.8.0peracionet aritmetike dhe shprehjet	
4.9. Konstantet dhe ndrvshoret	120
4.9.1. Konstantet	
4.9.2. Ndryshoret	
4.9.2.1. Tipi i ndryshores	
4.9.2.2. Vendosia e vlerës në ndrvshore	
4.10. Deklarimet (teknike) për vendosjen e të dhënave në program	123
4.11. Shprehjet krahasuese	126
4.11.1. Struktura për përzgjedhje nga dy mundësi	127
4.11.2. Shkruarja e programeve me strukturë për përzgjedhje	129
4.12. Struktura për përsëritie në cikël derisa të plotësohet kushti	
4.12.1. Shkruarja e programeve të thjeshta me struktura për përsërit	ie
deri në plotësimin e kushtit	
TË PËRSËRISIM! TË USHTROJMË!	138
5. Hvrie në ueb-ditarët (blogiet)	141
5.1. Koncepti për blogun dhe zbatimi i tij	143
5.2. Krijimi i blogut	
5.2.1. Hapat për krijimin e një blogu	145
5.2.2. Rregullimi dhe përmbajtja e blogut	147
5.2.3. Shtimi i figurës (imazhit) dhe videos në blog	
5.2.4. Shtimi i i lingeve në blog	149
5.2.5. Ndrvshimi i temës/pamies së bloqut	
5.3. Komentimi i përmbajtjes së blogut	152
TË PËRSËRISIM! TË USHTROJMË!	154
	157
	160
	100

## Programi për llogaritje tabelare

Hyrje në programin për llogaritje tabelare Programi për llogaritje tabelare – Microsoft Office Excel 2016 Puna me të dhëna në tabelë Rregullimi i tabelës Formatimi i tabelës Plotësimi automatik i të dhënave në tabelë Formulat dhe funksionet në programin për llogaritje tabelare Sortimi i të dhënave Krijimi i grafikut

TË PËRSËRISIM! TË USHTROJMË!

Action	Shortcut	Туре	Action	Shortcut	Туре	Action	Shortcut	Туре

## 1. Hyrje në programin për llogaritje tabelare

#### Çfarë do të mësojmë?

Të krijojmë tabela me tipe të ndryshme të të dhënave dhe një seri të llogaritjeve duke përdorur formula dhe funksione, të rregulluara me mjete për formatim dhe të paraqitura grafikisht nga grafikët.

Njerëzit në jetën e përditshme shpesh përballen me të dhëna të ndryshme që vendosen në tabelë. Për shembull: mësuesit në shkollë krijojnë tabela për të ndjekur dhe regjistruar arritjet (suksesin) e nxënësve për ndonjë lëndë, të jenë në gjendje të llogaritin suksesin mesatar të nxënësit dhe klasën në përgjithësi, të jenë në gjendje të bëjnë krahasime për arritjet gjatë tremujorit, për të krijuar një orar etj. Personeli i bibliotekës së shkollës mund të krijoj një tabelë me të dhënat e librave që janë në dispozicion të shkollës. Nxënësit mund të përdorin këtë program për qëllime të ndryshme shkollore dhe jashtëshkollore, si për shembull, krijimi i një tabele me të dhëna të nxënësve të klasës, krijimi i një libri telefonik, krijimi i tabelës për rregullsinë në shkollë, pasqyra e rregullsisë së detyrave të shtëpisë, mbledhja dhe përpunimi i të dhënave për një detyrëprojekt, paraqitja grafike e të dhënave të mbledhura etj. Prindërit mund të përdorin këtë program për të ruajtur evidencën për të ardhurat e familjes, shpenzimet mujore etj.

Prandaj, qëllimi i temës "Programi për llogaritje tabelare **MS Office Excel 2016**" është puna me të dhëna tabelare, realizimi i llogaritjeve sipas funksioneve të caktuara dhe formulave të krijuara, si dhe paraqitja grafike e të dhënave.

Nxënësit që mësojnë këtë temë do të jenë në gjendje të realizojnë:

- krijimin e një dokumenti tabelar;
- futjen e të dhënave të formateve të ndryshme;
- rregullimin/formatimin e tabelës;
- kryerjen e llogaritjeve duke përdorur formulat dhe funksionet;
- sortimin (klasifikimin) e të dhënave në tabelë;
- filtrimin e të dhënave në tabelë;
- krijimin e grafikëve të llojeve të ndryshme për prezantimin e të dhënave. Përveç përvetësimit teorik të koncepteve dhe materialit në tërësi, vëmendje të veçantë do t'i kushtohet punës praktike. Për këtë qëllim, duhet të instalohet paketa e zyrës Microsoft Office 2016.



Figura 1: Ikona e Microsoft Excel 2016



#### Termat kryesor!

librat e punës, fleta e punës, tabela, kolona, rreshti, qeliza, qelizat e njëpasnjëshme/ jo të njëpasnjëshme, qeliza aktive, formula, funksioni, grafiku.

### 1.1. Programi për llogaritje tabelare – Microsoft Office Excel 2016

#### Çfarë do të mësojmë?

Termin tabelë e njohëm në vitin e kaluar shkollor gjatë mësimit të temës "Programi për përpunimin e tekstit". Tabela paraqet një grumbull të dhënash të organizuara nga pjesa vertikale- të quajtura kolona dhe pjesa horizontale të quajtura rreshta. Kolonat dhe rreshtat priten, dhe prerja e tyre quhet qelizë.

Aplikacioni **Microsoft Office Excel 2016** është pjesë e paketit të zyrave të Microsoft Office 2016 dhe mundëson paraqitjen e të dhënave në mënyrë tabelare, rregullimin e tabelave, kryerjen e llogaritjeve tabelare dhe paraqitjen grafike të të dhënave, krijimin e raporteve profesionale dhe të rishikimit etj.

#### 1.1.1. Startimi i programit. Elementet e dritares së punës

Për të krijuar një dokument tabelar, startoni me programin Microsoft Office Excel 2016 përmes menysë startuese dhe zgjedhim ikonën e duhur që përfaqëson programin. Pastaj, paraqitet dritarja si në vijim:

Excel	Search for online templates			Informatika Informatika computers.doms.ns@gmail.com Switch.account
Recent Today New Microsoft Excel Work Desitop	A 8 C 1 2 3 4 5 6 7 Blank workbook	B         C         D           1         Your Company Name         1           3         Annual         1           4         Annual         1           5         Annual         1           4         Company Name         1           5         Annual         1           6         Company Name         1           7         Annual         1           8         Annual         1           9         Annual         1  <	Image: Second Pressure Tracker         0           3         Blood Pressure Tracker           4         Trge time prover           5         Griphman false           6         Griphman false           8         Griphman false           8         Blood Pressure Tracker	A Expense Report
	Loan Amortization	Personal Monthly Budget	Image: constraint of the state of	Time Card

Figura 1: Startimi i dokumentit të ri në Excel 2016

Kjo dritare është e ndarë në dy pjesë. Në anën e majtë të dritares shfaqet lista e dokumenteve që përdoruesi së fundmi i ka hapur dhe kanë vetinë e lidhjeve (linqeve), pra duke klikuar në emrin e dokumentit, ky dokument aktivizohet menjëherë. Në anën e djathtë të dritares, përdoruesi mund të hap një Blank Workbook, përkatësisht një dokument të ri ose të zbrazët të punës, ose të zgjedh një model të gatshëm (Template) të një dokumenti për përpunim ose ripërpunim.



Çfarë janë dokumentet? Si quhen dokumentet? Çfarë është karakteristikë e tyre?

Dokumentet e krijuara në programin MS Excel 2016 quhen librat e punës (Workbook) dhe marrin prapashtesën.xlsx. Çdo fletore pune përbëhet nga fletat e punës (Worksheet). Çdo fletë pune përbëhet nga tabela. Pjesët vertikale të tabelës quhen kolona dhe shënohen me shkronja (A, B, C.....Z, AA, AB, AC...), dhe pjesët horizontale quhen rreshta dhe shënohen me numra (1,2,3...). Qeliza është vendi ku futen të dhënat. Çdo qelizë ka një emër që përbëhet nga shkronja e kolonës dhe numri i rreshtit në të cilin ndodhet. Për shembull A1, B5, D16, etj. Çdo fletë pune përbëhet nga 256 kolona dhe 65 536 rreshta.

Gjatë përzgjedhjes së **Blank Workbook** aktivizohet dritarja e punës të programit **Microsoft Office Excel 2016**. Në figurën vijuese janë paraqitur pjesët e dokumentit të punës së programit.





1. **Quick Access Toolbar** – është shirit për qasje të shpejtë me komandat më të përdorura. Mund të modifikohet sipas nevojave të përdoruesit.

2. Shiriti me meny (Menu Bar) – është shirit menyje e që është një grupim i komandave sipas detyrave të tyre, në fakt, një shirit për komanda për kartën e duhur.

3. Shiriti me mjete (Toolbar) – paraqitet ose ndryshon në varësi të menysë që është përzgjedhur.

4. Adresa e qelizës aktive (Name Box) – është fushë për paraqitjen e emrit të qelizës së zgjedhur, pra tregon qelizën aktive.

5. Formula Bar – është shirit për futjen e formulave dhe funksioneve, duke shfaqur të dhëna ose përmbajtje të qelizës aktive.

6. **Qeliza aktive** – është qeliza në të cilën përdoruesi fut të dhëna.

7. Sheet Tabs – është shirit i fletës së punës.

8. **Scroll Bar** – janë shirita për të lëvizuar nëpër dritaren e punës: horizontale dhe vertikale.

9. Shiriti i statutit (Status Bar) – gjendet në fund të dritares dhe mund të përshtatet për paraqitjen e më shumë opsioneve, shumica e të cilave i japin përdoruesit informacion në lidhje me listën aktuale.

10. **Menu File** – i përmban urdhërat (komandat) themelor për punë me dokumentin në Excel.



#### 1.1.2 Puna me dokumente në Excel

Procedura për të punuar me dokumente në Excel është saktësisht e njëjtë si në programet e tjera të paketit për zyra Microsoft Office, përkatësisht të gjitha jepen përmes menysë File. Por, le të kujtojmë komandat dhe operacionet themelore me dokumente:

Aktiviteti/operacioni	Kuptimi i aktivitetit/operacionit
New	Hapja e dokumentit të ri
Open	Hapja e dokumentit ekzistues
Save	Ruajtja e dokumentit
Save As	Për krijimin e kopjes së re nga dokumenti në një lokacion tjetër ose me emër tjetër
Print	Shtypja e dokumentit
Share	Për ndarjen e dokumentit me më shumë shfrytëzues, ose për shembull mund të dërgohet përmes e-mail-it
Export	Për ndryshimin e llojit të dokumentit ose konvertimin e dokumentit në formatin .pdf ose .xps
Close	Për mbylljen e dokumentit

Tabela 1: Aktivitete/operacione me dokumentin e punës në Excel 2016

## Shënim!

Për prerjen e kolonave dhe rreshtave në këtë libër shkollor, do të përdorim fjalën qelizë. Megjithëse, në literatura të ndryshme mund të gjenden edhe emrin qelizë, edhe me emrin celulë.

#### 1.1.3 Puna me elementet themelore të fletës së punës

Dritarja e mëposhtme tregon elementet themelore të fletës së punës: tabela, rreshti, kolona dhe qeliza:



Figura 3: Elementet themelore të dritares së punës

Për të filluar futjen e të dhënave, së pari zgjedhim qelizën dhe i futim të dhënat në të. Qeliza që është selektuar dhe e përgatitur për futjen e të dhënave quhet qelizë aktive. Për shembull:



Krijoni një folder (dosje) në desktopin e kompjuterit tuaj. E emërtoni me emrin dhe mbiemrin tuaj. Në këtë folder do të ruani punët tuaja praktike.

Të hapet një dokument i ri pune.

Klikoni në qelizën AI dhe futni të dhënat: ushtrimi I. Klikoni në qelizën A2 dhe futni të dhënat: emrin tuaj. Klikoni në qelizën B2 dhe vendosni klasën tuaj. Ruani dokumentin me emrin "**UshtrimiI.xlsx**" në folderin që është krijuar në sipërfaqen e punës.

Kur punojmë me elementet në fletën e punës, gjithmonë duhet t'i selektojmë ose t'i shënojmë të njëjtat. Selektimi (përzgjedhja) ose shënimi do të thotë të tregoni se në cilat elemente do të kryejmë aktivitete, të tilla si: përzgjedhja e fontit, madhësia e fontit, ngjyra e fontit, kornizat, etj., si dhe kopjimi, prerja, fshirja, etj.

**Një qelizë** mund të selektohet duke klikuar mbi të. Në varësi të nevojës, mund të selektojmë më shumë qeliza në të njëjtën kohë. **Qelizat fqinje** i selektojmë ashtu që klikojmë në qelizën e parë duke klikuar në qelizën që dëshirojmë të selektohet, duke mbajtur tastin **Shift** në tastierë dhe klikohet në qelizën e fundit ose klikohet me maus në qelizën e parë dhe e tërhjekim zvarrë mausin deri në qelizën e fundit. Në këtë mënyrë është selektuar rangu i qelizave.

**Rangu i qelizave** paraqet tërësinë e qelizave të njëpasnjëshme ose qelizave fqinje, për shembull: A3: E11, dhe paraqitet si në vijim.



Figura 4: Selektimi i qelizave të njëpasnjëshme/fqinje

**Qelizat jo të njëpasnjëshme** ose **jo fqinje**, mund t'i selektojmë duke klikuar në qelizën e caktuar, pastaj e mbajmë të shtypur tastin **Ctrl** nga tastiera, dhe pastaj klikojmë në qelizën tjetër etj.



Figura 5: Selektimi i qelizave jo fqinje

Për lëvizjen nëpër dokumentin e punës do të përdorim shiritat lëvizës (Scrool Bars) të cilat gjenden në anën e djathtë dhe në pjesën e poshtme të dritares së punës, si dhe butonat në tastierë.

Tasti në tastierë	Lëvizja
Enter ose ↓	Një qelizë më poshtë
$\uparrow$	Një qelizë më lartë
÷	Një qelizë në të majtë
$\rightarrow$	Një qelizë në të djathtë
Ctrl + ←	Qeliza e parë e rreshtit
Ctrl + →	Qeliza e fundit e rreshtit
Home	Fillimi i rreshtit
End	Fundi i rreshtit
Ctrl + Home	Fillimi i tabelës
Ctrl + End	Fundi i tabelës

Tabela 2: Lëvizja nëpër dritaren e punës

#### Ushtrimi 1

Në dokumentin e ushtrimit 1, në fletën e punës Sheet1 të selektohen qelizat B4:F8 dhe me përdorim të mjetit Fill Color, nga menyja Home, të ngjyrosen me ngjyrë të kaltër. Fletën e punës e emërtoni: fqinjë. Të ruhen ndryshimet që janë bërë në dokument.

Në të njëjtin dokument shtoni fletë të re të punës. E emërtoni fletën e punës në- jo fqinjë. Selektoni qelizat: A2, C2, E2, G2, B3, D3, F3, A4, C4, E4, G4, B5, D5, F5, A6, C6, E6, G6, B7, D7, F7. Ndryshimet të bëhen në dokument dhe të ruhen.



Programi **Microsoft Office Excel 2016** përdoret për paraqitjen e të dhënave në tabelë, për kryerjen e llogaritjeve dhe paraqitjes grafike të të dhënave. Dokumentet që krijohen në Excel 2016 quhen fletore pune (Workbook) dhe kanë prapashtesën .xlsx. Fletoret e punës ose dokumentet përmbajnë fleta pune (Sheet1,Sheet2...). Fletat e punës përmbajnë tabela. Tabela është grumbull i të dhënave që janë të organizuara në pjesën vertikale të cilat quhen kolona dhe pjesa horizontale të cilat quhen rreshta. Çdo kolonë dhe rresht pritet (kryqëzohet), dhe prerja e tyre quhet qelizë ose celulë. Aktivitetet ose operacionet më të zakonshme me një dokument pune janë: hapja e dokumentit të ri, hapja e dokumentit ekzistues, ruajtja e dokumentit dhe shtypja e dokumentit.

## Pyetje

- 1. Pse përdoret programi Excel 2016?
- 2. Si quhen dokumentet e krijuara në programin për llogaritje tabelare?
- 3. Cilët janë elementet e dritares së punës në programin për llogaritje tabelare?
- 4. Si janë shënuar pjesët horizontale në dokumentin aktiv në programin për llogaritje tabelare?
- 5. Si quhet prerja e kolonës dhe rreshtit në programin për llogaritje tabelare?

#### 1.2 Puna me të dhëna në tabelë

## Shënim!

Të dhënat alfabetike dhe të dhënat alfa-numerike renditen në të majtë të qelizës, ndërsa të dhënat numerike renditen në të djathtë të qelizës. Sigurisht, me mjetet e formatimit mund t'i renditim sipas dëshirës.

Për të krijuar një tabelë në programin për llogaritje tabelare, së pari duhet të fusim të dhëna. Të dhënat në tabelë futen në qelizën aktive dhe pastaj vazhdohet me vendosjen e të dhënave, përdoren tastet në tastierë për të lëvizur nëpër dritaren e punës që e kemi paraqitur në njësinë e mëparshme.

Për shembull, për të krijuar një listë të nxënësve dhe lartësinë e tyre duhet të kryejmë hapat e mëposhtëm:

1. klikoni në qelizën A1 dhe futni emrin e nxënësit;

2. shtypni Enter në tastierë për të kaluar në qelizën A2;

3. vendosni një emër tjetër dhe përsëritni procedurën derisa të vendosni dhjetë (10) emra;

4. kur të vendosni emrin e fundit, klikoni në butonin **Tab** në tastierë, për të lëvizur, një qelizë në të djathtë. Vendosni lartësinë e nxënësit me numër;

5. me ndihmën e butonave për lëvizje nëpër dokumentin e punës, vazhdoni të futni të dhënat për nxënësit tjerë në listë;

6. ruani dokumentin në folderin (dosjen) të sipërfaqes së punës me emrin "**Ushtrimi2.xlsx**".

A	5	•	×	$\checkmark f_x$	Jovana
4	A	В	С	D	E
1	Ana		134		
2	Marko		130		
3	Safet		145		
4	Drita	Tab	133		
5	Jovana		131		
6					
7	25				
8		Enter			
9					

Figura 1: Vendosja e të dhënave në tabelë

Nga detyra praktike mund të vërejmë se të dhënat që i vendosim në qelizën aktive paraqiten edhe në **Formula Bar**, përkatësisht në **shiritin e formulave**. Me klikim në çfarëdo qelize, e cila përmban të dhëna, adresa e qelizës paraqitet në **Name Box**, ndërsa, përmbajtja e qelizës paraqitet në **Formula Bar**.

#### 1.2.1 Tipet e të dhënave

Të dhënat që i futim në tabelë mund të jenë **alfabetike**, **numerike** dhe **alfanumerike**. Të dhënat alfabetike përmbajnë vetëm shkronja, përkatësisht tekst, të dhënat numerike janë vargje të numrave, ndërsa, të dhënat alfa-numerike janë kombinim të shkronjave/tekstit dhe numrave.

#### Të dhënat alfabetike dhe alfa-numerike

Për vendosjen e të dhënave tekstuale **(alfabetike)** dhe kombinimin e tekstit dhe numrave **(alfa-numerike)** do ta realizojmë detyrën vijuese. Në dokumentin **"Ushtrime2.xlsx"**, të shtojmë fletë të re të punës, në të cilën do të krijojmë tabelë: të dhënat themelore për nxënësit. Riemërtoni fletën e punës me emrin **"Të dhëna"**.

1. Në qelizat A1, B1, C1, D1, E1, F1 dhe G1 do të krijojmë qeliza me titull:

Emri, mbiemri, adresa, vite, datëlindja dhe ora.

2. Duke përdorur butonët për lëvizje nëpër dokumentin e punës, vendosni të dhëna në kolonat: emri, mbiemri, adresa.

	D2	+ (o	fx.				
1	A	В	С	D	E	F	
1	Emri	Mbiemri	Adresa	Vite	Datëlindja	Ora	
2	Mila	Arsoska	Ivo Ribar Lola 143				
3	Arta	Emurli	Andrea Risoski 81				
4	Mirvet	Sela	Jovan Onçeski 24				
5	Selma	Bajrami	Bllagoja Toska 1/23				
6	Petar	Jovanoski	Ilirija 33-12/1				
7	Stefan	Milev	Braga Simoski 10				
8	Jana	Ristoska	KRD 22-1/3				
9							

Figura 2: Vendosja e të dhënave alfabetike dhe alfa-numerike

#### Të dhënat numerike

Kur flasim për të dhëna numerike, mund të shënojmë se ekzistojnë tipe të **ndryshme të të dhënave numerike**, siç janë: numrat e plotë, numrat pozitivë dhe negativë, data, ora, valuta etj. Le të plotësojmë tabelën **"Të dhënat"**. Pasi të futni të gjitha të dhënat, ruani ndryshimet në dokument.

	G2	<b>-</b> (°	fx					
1	А	В	С	D	E	F	G	Н
1	Emri	Mbiemri	Adresa	Vite	Datëlindja	Ora		
2	Mila	Arsoska	Ivo Ribar Lola 143	13	15.03.2006	18:30		
3	Arta	Emurli	Andrea Risoski 81	14	18.02.2005	22:40		
4	Mirvet	Sela	Jovan Onçeski 24	13	22.08.2006	14:10		
5	Selma	Bajrami	Bllagoja Toska 1/23	12	29.09.2007	19:20		
6	Petar	Jovanoski	Ilirija 33-12/1	15	12.12.2004	17:22		
7	Stefan	Milev	Braqa Simoski 10	16	28.03.2003	03:15		
8	Jana	Ristoska	KRD 22-1/3	14	03.03.2005	06:25		
-								

Figura 3: Vendosja e të dhënave numerike

Nga këtu, mund të vërejmë se kur vendosim një datë, ditën e ndajmë nga muaji me një pikë (.). Gjithashtu, muajin nga viti e ndajmë me një pikë. Kur futni një datë, gjithashtu mund të përdorim vijën e pjerrët (/) ose thjesht një vijë (-). Exceli më pas i njeh këto të dhëna si data. Kur e shënojmë orën, i ndajmë orët nga minutat dhe nga sekondat me shenjën me dy pika (:).

#### 1.2.2 Formati i të dhënave numerike

Kur i vendosim të dhënat në tabelë, mund të zgjedhim një format të dëshiruar. Përzgjedhja e formatit të të dhënave kryhet si në vijim:

1. selektohen të dhënat, formatin e të cilave dëshirojmë ta ndryshojmë;

2. klikoni me anën e djathtë të mausit në qelizat e selektuara dhe zgjedhim opsionin **Format Cells**.

Pastaj paraqitet dritarja vijuese:

	D2	- (°	<i>f</i> * 13					
	A	В	C	D		E	F	G
1	Emri	Mbiemri	Adresa	Vite	Da	tëlindja	Ora	
2	Mila	Arsoska	Ivo Ribar Lola 143	1	13 15	.03.2006	18:30	
3	Arta	Emurli	Andrea Risoski 81	1	14 18	.02.2005	22:40	
4	Mirvet	Sela	Jovan Onçeski 24	1	13 22	.08.2006	14:10	
5	Selma	Bajrami	Bllagoja Toska 1/23		Calibr	i * 11 *	A . \$ -	% , 🍼
6	Petar	Jovanoski	Ilirija 33-12/1		10		A - A - +	.0 .00
7	Stefan	Milev	Braqa Simoski 10	Ļ	0 20	.00.2000	×	00 →.0 🖽
8	Jana	Ristoska	KRD 22-1/3	5	V			
9				-	00	cu <u>r</u>		
10			Me klikim të djathtë	ë		Сору		
11					<b>B</b> .	<u>P</u> aste		
12						Paste <u>Special</u>	<b></b>	
13						Insert		
14						Delete		
15						Clear Conter	nts	
16						Filter		*
17						Cont		
18						3 <u>0</u> m		
19						Insert Co <u>m</u> m	ent	
20			Zgjedhja e opsionit	$\rightarrow$	<b>.</b>	Format Cells.		
21						Pic <u>k</u> From Dr	op-down List	
22						Name a Rang	je	
23						Hyperlink		
20018								

Figura 4: Zgjedhja e opsionit Format Cells

Pas zgjedhjes së opsionit Format Cells, paraqitet një dritare, ku zgjedhim menynë **Number**, dhe pastaj zgjedhim kategoritë:

			Forr	nat Cel	ls	?	×	
Number	Alignment	Font	Border	Fill	Protection			
Category: General Number	^	Sample						
Ceneral Number Currency Accounting Date Time Percentage Fraction Scientific Text Special Custom		General format cells have no specific number format.						
					OK	Can	cel	

Figura 5: Dritarja e Format Cells

- 1. **General** format i përgjithshëm.
- 2. Number numrat e plotë dhe dhjetorë, numrat pozitivë dhe negativë.
- 3. Currency formati i valutës: denarë, euro, dollar, lira, paund, etj.
- 4. Accounting i paraqet shenjat e valutës në anën e majtë të vlerës dhe paraqet një vizë (-) gjatë vlerës zero (0).
- 5. **Date** formatet e datave.
- 6. Time Format i kohës
- 7. Percentage paraqitja e formatit të përqindjes.
- 8. Fraction formati i thyesave.
- 9. Scientific e paraqet një numër në format eksponencial.
- 10. **Text** vargjet e tekstit.
- 11. **Special** formate speciale /të veçanta.
- 12. **Custom** zgjedhja e formateve shtesë të të dhënave.

Përveç procedurës së treguar për zgjedhjen e një formati të të dhënave numerike, gjithashtu mund të përdorim menynë Home dhe opsionin Number Format, siç tregohet në imazhin vijues:

	8 5· C.	÷				New Microsoft Offi	ce Exce	l Worksheet - Excel (Product Ac	tivation Failed)
	File Home	Insert F	age Layout Formulas	Data	Review V	liew Team	Q Tel	I me what you want to do	
P	Cut	Calibri	• 11 • A A	. = =	Ø).	Wrap Text	[	General	nditional Form
	<ul> <li>Format</li> <li>Clipboard</li> </ul>	Painter G	Font	5	Aligni	ment	ra	Number	matting * Tabl Styles
D	2 *	IX V	<i>f</i> = 13					Number Format	
	A	В	c	D	E	F	G	Choose the format for your o such as percentage, currency	cells, y, date K
1	Emri	Mbiemri	Adresa	Vite	Datëlindja	Ora		or time.	-
2	Mila	Arsoska	Ivo Ribar Lola 143	13	15.03.2006	18:30		7 Tell me more	
3	Arta	Emurli	Andrea Risoski 81	14	18.02.2005	22:40		here and her	
4	Mirvet	Sela	Jovan Onçeski 24	13	22.08.2006	5 14:10			
5	Selma	Bajrami	Bllagoja Toska 1/2	3 12	29.09.2007	19:20			
6	Petar	Jovanoski	Ilirija 33-12/1	15	12.12.2004	17:22			
7	Stefan	Milev	Braqa Simoski 10	16	28.03.2003	03:15			
8	Jana	Ristoska	KRD 22-1/3	14	03.03.2005	06:25			
-	Contraction of the second s								

Figura 6: Zgjedhja e formatit të të dhënave përmes menysë Home

Le të zgjedhim formatin e datës: dd.mm.yy dhe formatin e kohës, përkatësisht orën në të cilën do të shfaqet ora dhe minutat, si dhe opsionin paradite/pasdite. Ruani ndryshimet në dokument.



#### Mbaj mend!

Kur krijoni tabelë në programin për llogaritje tabelare (spreadsheet), të dhënat futen në qelizën aktive. Ekzistojnë tipe të ndryshme të të dhënave: alfabetike, numerike dhe alfanumerike. Të dhënat numerike mund të kenë formate të ndryshme: numri i plotë, numri dhjetor, numri pozitiv dhe negativ, data, koha, valuta, etj. Formatet e të dhënave i zgjedhim përmes opsionit Format Cells kur klikojmë me të djathtën e mausit (miut) ose përmes opsionit Number Format nga menyja Home.



#### Pyetje

- 1. Në cilën qelizë futen të dhënat?
- 2. Cili buton ose butona klikohen pasi të futni të dhënat në qelizë?
- 3. Numro formate të ndryshme të të dhënave!



Formati eksponencial nënkupton zëvendësimin e një pjese të një numri me E + n, në të cilin E (eksponenti) shumëzuar me numrin e mëparshëm të 10 në shkallën e n-të. Për shembull, 2-decimale format shkencor tregon 12345678901 si 1.23E + 10, që është 1.23 herë 10 në shkallën e 10-të.

#### 1.2 Rregullimi i tabelës

Kur krijojmë një tabelë në programin për llogaritje tabelare në shikim të parë duket se gjithçka është në rregull, por këtë duhet ta kontrollojmë për të shikuar nëse kemi bërë një gabim teknik, nëse kemi futur të dhëna të pasakta, nëse mungon një rresht ose një kolonë në tabelë. Sigurisht, nuk do ta fshijmë të gjithë tabelën dhe të fillojmë të krijojmë tabelën përsëri, por duhet të korrigjojmë gabimet që janë bërë. Për të bërë ndryshime do të hapim "**Ushtrimi2.xlsx**".

#### 1.2.1 Ndryshimi dhe rregullimi të dhënave të futura

Nëse duam të ndryshojmë të dhënat në qelizën B4, pra të fusim të dhëna të reja në vend të atyre ekzistuese, thjeshtë klikojmë mbi atë qelizë dhe fillojmë të shkruajmë. Në atë moment në shiritin e statusit do të shkruhet **Ready**, që do të thotë se jemi në modin e futjes së të dhënave të reja. Për shembull: në këtë qelizë mund të korrigjohet mbiemri, figura si në vijim.

Gjithashtu, në qelizën B2 duhet të shtojmë një karakter, pra një shkronjë në mbiemër.

Pastaj, me mausin, klikojmë dy herë në qelizën B2, vendosim kursorin në vendin ku duhet të fusim një shkronjë dhe shkruajmë shkronjën me tastierë, në rastin tonë shkronjën "v". Pastaj në shiritin e statusit do të shkruhet **Edit**, që do të thotë se jemi në modin e ndryshimit të të dhënave.

B3	5 *	: × ×	J <sub>x</sub> Baj
1	A	В	С
1	Emri	Mbiemri	Adresa
2	Mila	Arsoska	Ivo Ribar Lola 143
3	Arta	Emurli	Andrea Risoski 81
4	Mirvet	Sela	Jovan Onçeski 24
5	Selma	Baj	Bllagoja Toska 1/23
6	Petar	Jovanoski	Ilirija 33-12/1
7	Stefan	Milev	Braqa Simoski 10
8	Jana	Ristoska	KRD 22-1/3
9			

Figura 1: Ndryshimi i të dhënës së qelizës

B	2 🔻	: × ✓ .	<i>f</i> <sub>x</sub> Arsoska	
	A	В	с	
1	Emri	Mbiemri	Adresa	١
2	Mila	Arsoska	Ivo Ribar Lola 143	
3	Arta	Emurli	Andrea Risoski 81	
4	Mirvet	Sela	Jovan Onçeski 24	
5	Selma	Baj	Bllagoja Toska 1/23	
6	Petar	Jovanoski	Ilirija 33-12/1	
7	Stefan	Milev	Braqa Simoski 10	
8	Jana	Ristoska	KRD 22-1/3	
9				

Figura 2: Shtimi i shenjës/shkronjës në një qelizë

#### Shënim!

Për të futur më shpejt më shumë se një kolonë ose rresht, selektoni më shumë rreshta/kolona, klikoni me të djathtën mbi to dhe zgjedhim opsionin Insert.

E dhëna ose të dhënat nga qelizat mund të fshihen duke selektuar fillimisht të dhënat të cilat duam t'i fshijmë, dhe pastaj klikohet në butonin **Delete** në tastierë.

#### 1.3.2 Aktivitete me kolonat dhe rreshtat

Në tabelën e krijuar mund të shtojmë dhe fshijmë kolona dhe rreshta, varësisht nga nevoja e përdoruesit. Procedura për **futjen e një kolone të re** është si më poshtë:

- 1. selektoni kolonën para së cilës duam të shtojmë një kolonë të re;
- 2. kliko me të djathtën (e mausit) në shkronjën e kolonës së shënuar;
- 3. e zgjedhim opsionin Insert.

	Clipboard	5	Font 5	1	Cali	ibri + 11 + ^*		
D	1 *	: × ✓ fs	Vite		B I = 🖄 - 🗛 -			
	A	В	c —		)	E	F	
1	Emri	Mbiemri	Adresa	Vite	x	Cu <u>t</u>	а	
2	Mila	Arsoska	Ivo Ribar Lola 143			Сору	1	
3	Arta	Emurli	Andrea Risoski 81		-	Paste Options:	2	
4	Mirvet	Sela	Jovan Onçeski 24			-	1	
5	Selma	Baj	Bllagoja Toska 1/23				1	
6	Petar	Jovanoski	Ilirija 33-12/1			Paste Special	1	
7	Stefan	Milev	Braqa Simoski 10		_	Insert	0	
8	Jana	Ristoska	KRD 22-1/3	/	*	Delete	0	
9			/					
10			/			Clear Contents	_	
11					* 0	Eormat Cells		
12						Column Width		
13						Hide		
14						Habida		
15						Unnide		

Figura 3: Krijimi (insertimi)i kolonës së re në tabelë

Procedura është e njëjtë për futjen e një rreshti në tabelë:

1. selektoni kolonën para së cilës duam të shtojmë një rresht të ri; 2. kliko me të djathtën (e mausit) në numrin e rreshtit të shënuar;

3. e zgjedhim opsionin **Insert**.

4	A	В	1	С	D	E	F
Emi	ri Mi	biemri	Adre	sa	Vite	Datëlindja	Ora
Calib	ri + 11 +	A A \$ * %	, 🚍	par Lola 143	13	15.03.2006	18:30
в	$T \equiv \Diamond \cdot A$	· · · · · · · · · · · · · · · ·		a Risoski 81	14	18.02.2005	22:40
1				. Onçeski 24	13	22.08.2006	14:10
	-	-	Bllag	oja Toska 1/23	12	29.09.2007	19:20
00	Lut	noski	Ilirija	33-12/1	15	12.12.2004	17:22
	Copy	1.1.1.1	1:-41.4		16	28.03.2003	03:15
ĥ	Paste Options:	kliko me te	djathi	en (e mausit	) 14	03.03.2005	06:25
	<b>n</b> 1	në numrin e	rresh	tit të shënua	r		
I	Paste Special						
1	nsert		Zgj	edhja e opsi	onit		
[	Delete		Ins	ert			
	Clear Co <u>n</u> tents						
<b>E</b>	ormat Cells						
1	Row Height						
	Hide						
		-					

Figura 4: Krijimi (insertimi) i rreshtit të ri në tabelë

Krijimi (insertimi) i një kolone dhe rreshti në një tabelë mund të bëhet gjithashtu përmes menysë Home dhe zgjedhjen e mjeteve të përshtatshme për futjen e një kolone dhe rreshti:



Mundohuni!

#### Figura 5: Krijimi (insertimi) i kolonës dhe rreshtit përmes menysë

Megenëse kemi mësuar procedurën për krijimin e një kolone dhe rreshti në tabelë, le të përpiqemi të fshijmë një kolonë ose rresht nëse e dimë që në vend të opsionit Insert duhet të zgjedhim Delete.

#### 1.3.3 Ndryshimi i dimensioneve të kolonave dhe rreshtave

Kur të vendosen të dhënat në qeliza, shpeshherë duhet të ndryshojmë dimensionet e kolonave dhe rreshtave. Procedura për ndryshimin arbitrar të dimensioneve të kolonës, përkatësisht rreshtit, tregohet në figurën vijuese.

Nga figura mund të shohim se së pozicionojmë treguesin pari midis shkronjave të kolonës (kur e ndryshojmë dimensionin e kolonës), përkatësisht ndërmjet numrave të rreshtave (kur e ndryshojmë dimensionin e rreshtit), paragitet një kursor, shtypim dhe zvarritim. Kur ndryshoni madhësinë e rreshtit, kursori ndryshohet për 90°, dhe procedura është e njëjtë.

	· ·		√ Jx	Adr	esa 🗸	Width: 17.	29 (1
A	A	1	3		c	D	
1	Emri	Mbiemri		Adresa		Vite	Dat
2	Mila	Arsoska		Ivo Riba	ar Lola 143	13	15.
3	Arta	Emurli		Andrea Risoski 81		14	18.
4	Mirvet	Sela	ela		Jovan Onçeski 24		22.
5	Selma	Baj	aj		Bllagoja Toska 1/23		29.
6	Petar	Jovanosk	ovanoski		Ilirija 33-12/1		12.
7	Stefan	Milev		Braqa S	imoski 10	16	28.
8	Jana	Ristoska		KRD 22-	1/3	14	03.
-	1	- 	~	£	Adros		
			B	JA	Aures	-	1
1	Emri	Mbi	Mbiemri			-	h
			emn		Adresa		
2	Mila	Arso	ska		Adresa Ivo Ribar I	Lola 143	-
2 3	Mila Arta	Arso	enn ska rli		Adresa Ivo Ribar I Andrea Ri	Lola 143 soski 81	-
23	Mila Arta	Arso Emu	oska rli		Adresa Ivo Ribar I Andrea Ri Jovan Ong	Lola 143 soski 81 ceski 24	
2 3	Mila Arta leight: 15.00 (	Arso Emu (20 pixels) Baj	oska Irli		Adresa Ivo Ribar I Andrea Ri Jovan Ong Bllagoja T	Lola 143 soski 81 çeski 24 oska 1/23	
2 3	Mila Arta Height: 15.00 ( Seuma Petar	Arso Emu (20 pixels) Baj Jova	noski		Adresa Ivo Ribar I Andrea Ri Jovan Ong Bllagoja T Ilirija 33-1	Lola 143 soski 81 ceski 24 oska 1/23 2/1	
2 3 5 6 7	Mila Arta Height: 15.00 ( Seuma Petar Stefan	Arso Emu (20 pixels) Baj Jova Mile	noski		Adresa Ivo Ribar I Andrea Ri Jovan Ong Bllagoja T Ilirija 33-1 Braqa Sim	Lola 143 soski 81 çeski 24 oska 1/2: .2/1 oski 10	

kolonës dhe rreshtit

Shënim!

Mund të ndryshoni dimensionet e kolonave/rreshtave në një mënyrë më të shpejtë duke klikuar dy herë në vijën pas shenjës së kolonës /rreshtit.

Hapni dokumentin "**Ushtrimi2.xlsx**" dhe ndryshoni dimensionet e kolonës ku ndodhet datëlindja dhe rreshti i përparmë. Pastaj, kryeni procedurën e ndryshimit të kolonës/rreshtit përmes opsionit Format nga menyja Home, me qëllim që të krijoni të njëjtat dimensione për të gjitha kolonat dhe rreshtat.

Por, kur përdoruesi dëshiron të zmadhoj ose zvogëloj kolonat dhe rreshtat për një vlerë specifike, atëherë përmes menysë Home, zgjedhim mjetin Format:



Detyrë



Figura 7: Ndryshimi i dimensioneve të rreshtit përmes opsionit Format nga menyja Home





Rregullimi dhe editimi i dokumentit është një procedurë korrigjimi. Për ta bërë këtë, mund të ndryshojmë të dhënat duke klikuar në qelizë dhe menjëherë të fillojmë me shkruarjen, ose duke klikuar dy herë dhe duke pozicionuar kursorin në vendin ku duam të bëjmë korrigjime. Përmes menysë Home mund të shtojmë kolona dhe rreshta, të fshijmë kolonë dhe rresht, si dhe të ndryshojmë dimensionet.



- 1. Cila është procedura për ndryshimin e të dhënave në një qelizë?
- 2. Si mund të ndryshohen dimensionet e kolonave dhe rreshtave: Theksoni procedurat!

## 1.4 Formatimi i tabelës

## Të rikujtojmë

Kujtohu se çfarë është formatimi! Cilat mjete përdoren më shpesh për formatimin? A mund të shpjegoni se për çfarë duhet të keni kujdes më shumë kur formatoni?

Pamja e qelizave, kolonave dhe rreshtave finalizohet duke formatuar pamjen e tabelës.

Formatimi i të dhënave është në fakt shtimi i efekteve që rritin pamjen e tabelës, duke mundësuar kështu dukshmëri më të madhe të të dhënave në të.

Formatimi i qelizave paraqet:

- ndryshimet e atributeve për fontin;
- njëjtësimi (barazimi) i të dhënave në qeliza;
- paraqitja e tekstit në një kënd;
- bashkimi dhe ndarja e qelizave;
- shtimi i kornizave dhe linjave në një ose më shumë qeliza;
- shtimi i ngjyrës në qelizë dhe shtimi i efekteve.

Por, përpara se të ndërmerret ndonjë veprim, qelizat duhet së pari të selektohen. Shumica e mjeteve të formatimit janë në menynë **Home**:

	• ♂ + ≠								Bo	ok1 - Excel				
File	1 Home	Insert Pa	ge Layout	Formulas	Data	Review	View	♀ Tell me	what you w	ant to do	_			
Paste 2 Clip	5 Cut § Copy → I Format Paint ¤board	Calibri B I S 3	4			≡ »·· ≡ •≣ •	ignment	ap Text 12 rge & Cente 13	r T	ral % * *	• <sup>00</sup> → 00 For 4 5 1	onditional 5 rmatting ™ 5 9	ormat as Table + yles 16	Cell tyles≁ 17
A1	-	× ✓	fx	0 /										1.
A	В	С	D	E	F	G	н	T	J	к	L	м	N	0
1														
2														
3														

Figura 1: Mjetet për formatim nga menyja Home

- 1. Menuja Home.
- 2. Komandat e kopjimit, prerjes, futjes dhe mjetet për formatim.
- 3. Stilet: të theksuara, të pjerrëta dhe të nënvizuara.
- 4. Përzgjedhja e fontit dhe madhësia e fontit.
- 5. Përzgjedhja e skajeve.
- 6. Përzgjedhja e ngjyrës së qelizës.
- 7. Përzgjedhja e ngjyrës së fontit.
- 8. Mjetet për zmadhimin/zvogëlimin e shkronjave.
- 9. Orientimi i të dhënave.
- 10. Vizatimi.
- 11. Ndarja e tekstit.

- 12. Bashkimi i qelizave dhe rregullimi në qendrës.
- 13. Zgjedhja e tipit të të dhënave.
- 14. Formatimi i kushtëzuar.
- 15. Formatimi i tabelës.
- 16. Stilet e qelizave.

Formatimi i tabelës, përveç me menynë **Home** mund të realizohet edhe me opsionin **Format Cells** pasi të keni selektuar më parë qelizat nga tabela:

Pumber     Alignment       Horizontal:     General       Vertical:     Bottom       Justify distribut       Text control       Wrap text       Shring to fit       Merge cells       Right-to-left       Jext direction:       Context	at Font	Indent:	at Cells	5 Protection	? × Orientation T t T ext → t t 0 © Degrees	<ol> <li>I. Zgjedhja e formatit të të dhënave.</li> <li>Rregullimi i të dhënave.</li> <li>Zgjedhja e fontit.</li> <li>Zgjedhja e kufijve dhe opsioneve për kufizimet</li> <li>Plotësimi i qeliave/qelizave me ngjyra.</li> <li>Mbrojtja e fletës së punës.</li> </ol>
					OK Cancel	

Figura 2: Formatimi i të dhënave përmes menysë Format Cells

Përmes tab-it Alignment përdoruesi mund të kryej renditjen e përmbajtjeve të qelizave, përkatësisht t'i rendit të dhënat në qelizë, ashtu që në Text Alignment zgjedhet renditja e të dhënave në mënyrë horizontale dhe vertikale, si dhe orientimi ose drejtimi i tekstit në qelizë.

Rregullime plotësuese kryhen edhe me zgjedhjen e opsioneve vijuese:

Wrap Text – paraqet tekstin i cili do të renditet në shumë rreshta.

Shrink To Fit – e zvogëlon tekstin me qëllim që të përfshihet në qelizë pa i ndryshuar dimensionet e qelizës.

Merge Cells – mundëson bashkimin e qelizave të selektuara.

Përmes zgjedhjes së tabit **Font**, zgjedhet forma e shkruarjes së shkronjave, stileve, madhësisë dhe efekteve tjera të cilat i përkasin fontit:

	Form	nat C	Cells	? ×
	Number Alignment Font Border	Fill	Protection	
	<u>F</u> ont:		Font style:	Size:
	Calibri		Regular	11
	Tr Calibri Light (Headings) Tr Calibri (Body) Tr Agency FB Tr Aharoni	^	Regular Italic Bold Bold Italic	8 9 10 11
	Tr Aldhabi	~	~	12
	<u>U</u> nderline:		<u>C</u> olor:	
	None	~		✓ ✓ <u>N</u> ormal font
	Effects		Preview	
	☐ Stri <u>k</u> ethrough ☐ Sup <u>e</u> rscript ☐ Su <u>b</u> script		AaBbC	CcYyZz
	This is a TrueType font. The same font will be	used	on both your printer an	id your screen.
			0	K Cancel
7	Figura 3: Zgjedhja	e r	njeteve për	fontin

Opsionet për fontin, stilin e fontit dhe madhësinë e fontit mund ta zgjedhim me menynë Home.

Në opsionin **Border**, zgjedhim stilin, trashësinë dhe ngjyrën e skajeve të qelizës së tabelës.

Number	Alignment	Font	Border	Fill	Protection				
Line		Pres	ets						
Style:									
None	· · · · · · ·	-							
			None	Outline	Inside				
		Bord	ler						
		-	3		L.				
	·····	-							
1				Text					
Color:									
Au	tomatic v				г				
		2				$\sim$			
The select above.	ted border styl	e can be a	pplied by cli	cking the	presets, pre	view dia	gram or	the but	tons

Figura 4: Zgjedhja e kufijve dhe opsioneve për kufizimet

#### Shënim!

Menyja Home i përmban mjetet për rregullim të skajeve të tabelës, si mundësi për qasje më të shpejtë dhe më të lehtë.

Number Backgro	Alignment und <u>C</u> olor: No Colo	Font	Border	Fill Patte	Protection	
Backgro	und <u>C</u> olor: No Colo	r		Patte	C-law	
	No Colo	r			rn Color:	
					Automatic	 ~
				Patte	rn Style:	
						~
Fill Fff	ects Mo	re Colors.				
1 In Chi	<u></u>	ine conorsi.				
c						
Sample						

Përmes opsionit Fill zgjedhet ngjyra për të mbushur qelizat:

Figura 5: Zgjedhja e ngjyrës për prapavijën e qelizës

## Detyrë

Hapni një fletë pune të re në fletën e parë të punës (Sheet1) krijoni një orar:

- në qelizat A1, B1, C1, D1, E1 shkruani emrat e ditëve të javës në të cilat i ndiqni mësimet;
- në qelizat poshtë qelizave të titullit, vendosni lëndët sipas orarit të klasës;
- shtoni një kolonë të re para kolonës e hënë dhe shënoni një numër rendor të orëve;
- shtoni një rresht të ri mbi qelizat e titullit;
- bashkoni gjashtë qelizat e para dhe shkruani titullin "Orari";

- qelizat e titullit në të cilat janë shkruar emrat e ditëve i bëni me fontin Calibri, madhësia katërmbëdhjetë (14), stilet e theksuara (bold) dhe të pjerrëta, ngjyra e shkronjave kaltër e errët dhe ngjyra e prapavijës (backgorund) të qelizave e kaltër e çelët;

- -tabelës i krijoni skajet (kufizimet);
- vendosni dimensionet e kolonave/rreshtave sipas nevojës;
- e ruani dokumentin në folderin e sipërfaqes së punës (desktop).

Formatimi i qelizave të gatshme gjithashtu mund të realizohet duke zgjedhur mjetin **Cells Styles** nga menyja **Home**, siç tregohet në figurë. Sidoqoftë, qelizat duhet të zgjidhen (selektohen) paraprakisht në mënyrë që stili i zgjedhur nga përdoruesi të zbatohet nga opsionet e ofruara:

				Informatika Infe	ormatika 🧏 Shar
	itional Format as atting ▼ Table ▼ St	Cell Insert Dele	× Inter Format	toSum • AZ • Sort & F ear • Filter • S	Find & elect *
Good, Bad and N	leutral				
Normal	Bad	Good	Neutral		
Data and Model					
Calculation	Check Cell	Explanatory	Input	Linked Cell	Note
Output	Warning Text				
Titles and Heading	ngs				
Heading 1	Heading 2	Heading 3	Heading 4	Title	Total
Themed Cell Sty	les				
20% - Accent1	20% - Accent2	20% - Accent3	20% - Accent4	20% - Accent5	20% - Accent6
40% - Accent1	40% - Accent2	40% - Accent3	40% - Accent4	40% - Accent5	40% - Accent6
60% - Accent1	60% - Accent2	60% - Accent3	60% - Accent4	60% - Accent5	60% - Accent6
Accent1	Accent2	Accent3	Accent4	Accent5	Accent6
Number Format					
Comma	Comma [0]	Currency	Currency [0]	Percent	
Comma           New Cell Styles           Merge Styles	Comma [0] e	Currency	Currency [0]	Percent	

Figura 6: Selektimi i stileve të qelizave

Përveç formateve të gatshme të qelizave, përdoruesi mund të zgjedhë një format të gatshëm të tabelës duke zgjedhur mjetin Format As Table nga menyja Home, të paraqitur në figurën vijuese:

I V	Format as Table •	Cell Styles *	e II Inser	t Delete Fo	rmat v Cl	utoSum * , II * S lear * F	AZY P Sort & Find & Filter * Select	2	
	Light		_		-				<b>F</b>
				-	-	_			
	=====								
	[								
1						=====	=====		
1	Medium								
		-							
						22222			
			==						
-									
-									
-									
-	Dark								
	Junk								
							=====		-
	Men New	Table St	de						-
+	*	Di unit I	La Ch. I						
_	1. New	Pivot I ab	ie style	2					1
4	1							1	D. D. L.

Figura 7: Selektimi i stileve të qelizave

#### Shënim!

Përveç formatimit standard të tabelës, ekziston edhe një formatim i veçantë, pra formatimi i kushtëzuar. Këtij formatimi i qasemi përmes menysë Home dhe opsionit Conditional Formatting. Provoni të përdorni këtë metodë formatimi në një nga tabelat e krijuara. Detyrë

Shtoni një fletë pune të re dhe krijoni një tabelë në të cilën do të vendosni të dhëna për fillimin dhe mbarimin e orëve dhe kohëzgjatjen e pushimeve ndërmjet orëve. Formatoni tabelën duke aplikuar në stilet e qelizave (**Cell Styles**) nga menyja **Home** ose duke zgjedhur një format të gatshëm të tabelës përmes opsionit **Format Table** përsëri nga menyja **Home**.



#### Mbaj mend!

Formatimi i një tabele nënkupton shtimin e efekteve në tabelë që përfshinë ndryshimin e mënyrës, formatit, orientimit dhe shtrirjes së të dhënave në qelizë. Kur formatoni një tabelë, përdoren më shpesh mjetet nga menyja **Home** ose përmes komandës **Format Cells**. Ekzistojnë edhe forma të gatshme të qelizave dhe tabelave dhe i zgjedhim ato përmes menysë **Home** dhe komandave **Cells Styles** dhe **Format As Table**.



- 1. Çfarë është formatimi?
- 2. Cili është ndryshimi ndërmjet formatimit dhe rregullimit të një tabele?
- 3. Cilat mjete përdoren më shpesh për formatimin e tabelës?
- 4. Si rregullohet shtrirja dhe orientimi i tekstit në qeliza?
- 5. Cilat opsione mund të zgjedhim përmes komandës Format Cells? A mund t'i qasemi atyre në ndonjë mënyrë tjetër? Shpjegoni!
- 6. Cila është procedura për ndarjen e një teksti?
- 7. Pse shërben komanda Merge Cells?
- 8. Pse shërben opsioni Cells Styles nga menyja Home?
- 9. A përdorni format të gatshëm të tabelës? Cila është procedura e përzgjedhjes?

## 1.5 Plotësimi automatik i të dhënave në tabelë

## Të rikujtojmë

A ju kujtohet se çfarë është përpunimi i të dhënave kompjuterike? Te të dhënat cilat janë përparësitë e përpunimit kompjuterik? A mund ta shpjegoni termin përpunim "automatik"?

Plotësimi automatik i të dhënave të tabelës është një mjet shumë i dobishëm i **Excel**-it. Ky mjet lejon që në vend se të dhënat të futen manualisht, mund të përdorim funksionin e Autofill për të plotësuar me të dhëna qelizat. Plotësimi automatik është një nga funksionet kryesore në përpunuesin e tabelës, prandaj të dhënat e dëshiruara futen në tabelë për më pak kohë, në krahasim me mënyrën manuale.

Në fakt, **Autofill** (plotësimi automatik) i **Microsoft Excel** na lejon të krijojmë tabela në mënyrë më efikase, duke na lejuar të mbushim më shpejt qelizat me një sërë të dhënash, jo vetëm me vlera të thjeshta numerike, por edhe me tipe tjera të të dhënave ose formulave. Përveç kësaj, është e mundur të krijoni lista individuale që do të lejojnë përdoruesin të mos humb kohë, çdo herë duke hyrë në një hyrje të re me të njëjtat vlera.

Por, përpara se të njohim funksionin dhe zbatueshmërinë e mbushjes automatike të qelizave, le të njohim kuptimet e formave të ndryshme të treguesve.

Treguesi	Kuptimi
+	Plotësimi automatik i qelizës
$\mathcal{V}_{\theta}$	Kopjimi i përmbajtjes së qelizës
4	Zhvendosja e përmbajtjes së qelizës
+++	Ndryshimi i dimensionit të kolonës
÷	Ndryshimi i dimensionit të rreshtit
Ŧ	Shënimi/mark-imi i kolonës
<b>→</b>	Shënimi/mark-imi i rreshtit
0	Shënimi/mark-imi i qelizës

Tabela 1: Forma të treguesit

Për të plotësuar qelizat automatikisht do të përdorim treguesin e mëposhtëm: +. Procedura për mbushjen automatike të qelizave është si më poshtë:

1. klikoni në ikonën që do ta shënojmë si qelizë aktive dhe vendosni të dhëna;

2. pasi të shënojmë të dhënat, shtypim Enter nga tastiera;

me mausin vijmë në këndin e poshtëm të djathtë të qelizës ku ndodhen të dhënat;

3. marrim tregues për mbushje automatike;

4. klikoni dhe tërhiqeni deri te qeliza në të cilën duam të kryejmë mbushjen automatike.

5. klikojmë dhe e tërhjekim deri te qeliza në të cilën duam të realizojmë plotësimin automatik.



Shënim!

Figura 1: Plotësimi automatik i qelizave

Në programin për llogaritje tabelare mund të krijohen lista, siç janë: ditët e javës, muajt e vitit, etj. Procedura për krijimin e listave bëhet përmes menysë File, Options, menynë Advance dhe zgjedhim Edit Custom List. Mundohuni të krijoni një listë të gatshme!



#### Detyrë

1. hapni një dokument të ri pune në Excel 2016;

2. në fletën e punës 1 (Sheet 1), në kolonën A, të krijohet një varg i numrave çift duke përdorur mbushjen automatike të qelizave, duke filluar nga numri dy (2) deri në numrin njëzet e dy (22);

3. në kolonën B, në të njëjtën fletë pune, krijoni një varg të numrave tek duke filluar nga numri tre (3), deri në njëzet e tre (23);

4. në kolonën C krijoni një vargme emrin Ana, ndërsa në kolonën D një varg me emrin Ana Maria;

5. në kolonën E krijoni një varg që do të ndryshoj për vlerën pesë (5) duke filluar nga zero (0);

6. në kolonën F plotësoni vargun 100, 99, 98 ..... 89.

## Mbaj mend!

Mbushja automatike e qelizave është një nga përparësitë e shumta të programit për llogaritje tabelare, pasi lejon të fusim të dhëna, të krijojmë vargje dhe të mbushim qelizat më lehtë dhe më shpejtë. Për të kryer mbushje automatike të qelizave përdorim treguesin vijues: +. Plotësimi automatik mund të bëhet jo vetëm me të dhëna që përmbajnë tekst, por edhe numra, formate të ndryshme të të dhënave, formula dhe funksione. Në programin për llogaritje tabelare, mund të krijohen lista të veçanta që përdoren për të mbushur automatikisht qelizat.

## 1.6 Formulat dhe funksionet në programin për llogaritje tabelare

Siç kemi theksuar në vet përkufizimin, programi Excel, përveç që siguron të dhëna tabelare dhe përmbledhëse, gjithashtu na lejon të kryejmë llogaritjet. Prandaj, qëllimi kryesor për zbatimin e formulave dhe funksioneve në programin për llogaritje tabelare është kryerja e llogaritjeve të shpejta dhe të sakta.

#### 1.6.1 Formulat

Formulat në programet për llogaritje tabelare krijohen për të realizuar llogaritje përmes operatorëve të ndryshëm matematikor. Excel 2016 përdor shenjat standarde për formulat të paraqitura në tabelën vijuese:

Shenja	Operacioni matematikor
+	Mbledhja
-	Zbritje
*	Shumëzim
/	Pjesëtim
^	Shkallë/eksponent

Tabela 1: Operatorët matematikor për kryerjen e llogaritjeve

## Shënim!

Mund të përdoren operatorët logjikë, si: <,>, =, <=,> =, <>, të cilët kthejnë një vlerë të vërtetë (True) ose të gabuar (False). Gjatë krijimit të formulave mund të përdoren kllapat.

Formulat dhe funksionet futen në qelizë nga tabela dhe gjithmonë fillojnë me shenjën "=".

Për shembull:

Për të llogaritur sa duhet të paguajmë në arkë në supermarket me përdorim të formulave; duhet të futen dhënat, dhe të realizohen llogaritjet:

- 1. klikojmë në qelizën D2 dhe shënojmë shenjën "=";
- 2. e klikojmë adresën e qelizës në të cilën gjendet sasia;
- 3. e shënojmë shenjën e shumëzimit dhe e klikojmë adresën e qelizës në të cilën është shënuar çmimi i produktit;
- 4. shtypet Enter nga tastiera

C2		*		× ✓	f <sub>x</sub>	*C2		
	A	В		С	D		E	1
1	Prodhimi	Sasia	IJ	Çmimi	Gjiths	ej		
2	Buka		3	25	=B2*C	2		
3	Jogurt		2	48	6			
4	Qumësht		2	43				
5	Sheqer		3	23				
6	Kafe		4	85				
7	Gjalpë		1	96				
8	1.1.1.1.1.1							

Figura 1: Llogaritja me aplikim të formulës

## Shënim!

Përveç shkruarjes së adresave të qelizave, adresën e qelizës në formulë mund ta shtojmë duke klikuar mbi të.

Nga detyra, vërejmë se kur vendosni një formulë në qelizën aktive, duke klikuar butonin Enter në tastierë në qelizën aktive, paraqitet rezultati dhe në shiritin e formulës (Formula Bar) shfaqet formula që kemi përdorur për gjetjen e rezultatit. Procesi i llogaritjes për të gjitha artikujt tjerë, realizohet duke i plotësuar automatikisht qelizat tjera.

Le të llogaritim shumën totale që duhet paguar!

1. Klikoni në D8 dhe shënoni shenjën "=".

2. Shënojmë adresat e qelizave nga D2: D7, duke shënuar shenjën e mbledhjes "+" ndërmjet qelizave të përzgjedhura.

3. Klikojmë Enter në tastierë dhe fitojmë rezultatin e përgjithshëm.

S	JM	•	× ✓	f <sub>x</sub> =D	2+D3+D4+I	D5+D6+D7	D	8	•	× ✓	<i>f</i> <sub>x</sub> =D2	+D3+D4+I	D5+D6+D7	
	A	В	с	D	E	F		A	В	с	D	E	F	
1	Prodhimi	Sasia	Çmimi	Gjithsej			1	Prodhimi	Sasia	Çmimi	Gjithsej			
2	Buka	Э	25	75			2	Buka	3	25	75			
3	Jogurt	2	48	96	5		3	Jogurt	2	48	96			
4	Qumësht	2	43	86	5]		4	Qumësht	2	43	86			
5	Sheqer	3	23	69	Ī		5	Sheqer	3	23	69			
6	Kafe	4	85	340	D		6	Kafe	4	85	340			
7	Gjalpë	1	96	96	5		7	Gjalpë	1	96	96			
8	and the second second			=D2+D3+l	04+D5+D6+	-D7	8	and the second second			762			
9							9							

Figura 2: Llogaritja e shumës së përgjithshme me aplikim të formulave

#### Shënim!

Gjatë krijimit të formulave nuk do të veprohet me vlera të qelizave, por me adresat e tyre, sepse në këtë mënyrë mund ndryshojmë automatikisht rezultatin, kur e ndryshojmë vlerën e të dhënave në qeliza.



D	2	•	$\times \checkmark f_x$	
	A	В	с	D
1	Х	Y	z=x*2-2*y	s=(x+y)/4
2	0	3	1	
3	1	6	i	
4	2	9	)	
5	3	12	2	
6	4	15	;	
7	5	18	3	
8	6	21	L	
9	7	24	L	
10	8	27	7	
11	9	30		
12	10	33		

#### 1.6.2 Funksionet

Funksionet janë formula të gatshme që mund të përdoren për të kryer llogaritjet e caktuara me të dhënat në tabelë. Funksionet kanë emrin e tyre dhe grupohen në kategori: matematikore, statistikore, logjike, etj. Funksionet shfrytëzojnë adresat e qelizave, rangun e qelizave ose qelizat jo fqinje.

Para funksionit shkruhet shenja barazi (=), pastaj shënohet emri i funksionit dhe në fund argumentet e funksionit. Argumentet e funksionit janë në të vërtetë emërtimet e qelizave që do të marrin pjesë në llogaritje.



Figura 3: Elementet e funksionit

Funksionet më të përdorura janë:

- SUM logarit shumën e vlerave numerike në një grup qelizash.
- MIN gjenë vlerën më të vogël numerike nga një grup qelizash.
- MAX gjenë vlerën më të madhe numerike nga një grup qelizash.
- AVERAGE llogarit mesataren e vlerave numerike në një grup qelizash.

COUNT – gjenë numrin e shënimeve numerike në qelizat e zgjedhura.

**COUNTIF** – gjenë numrin e të dhënave në qelizat e selektuara nëse plotësohet një kusht i caktuar.

Procedura për vendosjen e një funksioni fillon me selektimin e qelizës në të cilën do të kryhet llogaritja, dhe pastaj përmes menysë **Formulas** dhe komandës **Insert Function** kalojmë në hapin për vendosjen e funksionit.



Figura 4: Procedura për vendosjen e funksioneve

Pastaj, paraqitet figura vijuese, nga e cila zgjedhim kategorinë dhe funksionin:

	Insert Functio	n	? ×
Search for a function:			
Type a brief descript click Go	ion of what you want to	do and then	<u>G</u> 0
Or select a <u>c</u> ategory:	Most Recently Used	~	
Select a function:			
SUM AVERAGE IF HYPERLINK COUNT MAX SIN			~
SUM(number1;numl Adds all the number:	per2;) ; in a range of cells.		
Help on this function		ОК	Cancel

Figura 5: Zgjedhja e funksionit

Për të kryer llogaritje me ndihmën e funksionit të përzgjedhur, shtohen argumente si në dritaren në vijim:

	Function Arguments	?	×
SUM Number1 Number2	Image: Second se		
Adds all the numbers in a	= range of cells. Number1: number1;number2; are 1 to 255 numbers to sum. Logica text are ignored in cells, included if typed as arguments.	l valu	es and
Formula result = <u>Help on this function</u>	ОК	Car	ncel

Figura 6: Zgjedhja e argumentit të funksionit
Me klikim në butonin OK nga dritarja do të fitohet rezultati i llogaritjes.

Por, nganjëherë në qelizën ku duhet të paraqitet rezultati i formulës ose funksionit, paraqiten simbolet e gabimit: #####, # DIV / 0!, #NAME?, #REF!, #VALUE!. Këto quhen gjithashtu mesazhe gabimi në formulë ose funksion (**Error message**).

Porositë për gabime	Kuptimi
#####	Kolona është e ngushtë për të paraqitur rezultatin
# DIV / 0!	Pjesëtimi me zero nuk është i mundur
#NAME?	Në formulë ka gabim në adresën e shënuar
#REF!	Në formulë ka adresë të qelizës që është fshirë
#VALUE!	Në formulë një nga qelizat përmban tekst

Tabela 1: Operatorët matematikor për kryerjen e llogaritjeve

# Detyrë

#### Të krijohet tabela me programin për llogaritje tabelare si në figurën në vijim:

11	3 • : × ✓	fx						
	Α	В	С	D	E	F	G	н
1						1.11		1.5
2		Num	nri i librav	e të lexuar	sipas mua	jve		
3	Emri dhe mbiemri i nxënësit	Janar	Shkurt	Mars	Prill	Maj	Gjithsej libra të lexuar	
4	Ana Mihajloska	4	3	2	1	0		
5	Sara Etemi	5	2	2	1	1		
6	Arta Tahiri	4	3	1	2	1		
7	Jovan Spiroski	3	1	2	1	1		
8	Driton Emini	6	1	3	2	0		
9								
10	Gjithsej libra të lexuar sipas muaive						a	
11	Libra më shumë të lexuar						( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( )	
12	Libra më pak të lexuar							
13	Vlera mesatare të librave të lexuar		2			0		
14	Sa nxënës nuk kanë lexuar asnjë libër					8		
15								

Realizoni llogaritjet e kërkuara, duke përdorur funksionet: SUM, MIN, MAX, AVERAGE, COUNT dhe COUNTIF.



Formulat në programet për llogaritje tabelare janë shprehje matematikore të krijuara nga të dhënat dhe operatorët e ndryshëm matematikor. Formulat nuk kanë të bëjnë me të dhënat në qeliza, por me adresat e tyre. Funksionet janë formula të gatshme që mund të përdoren për të kryer llogaritjet e caktuara me të dhënat në tabelë. Funksionet kanë emrin e tyre dhe grupohen sipas kategorive: matematikore, logjike, statistikore, etj.



- 1. Çfarë janë formulat dhe cilat janë funksionet?
- 2. Përshkruani ndryshimin në mes të formulave dhe funksioneve!
- 3. Cilët funksione përdoren më shpesh?
- 4. Hulumtoni, numëroni dhe theksoni funksione të tjera në Excel përveç atyre që janë mësuar në njësinë mësimore!
- 5. Shkruani sintaksën e funksionit që llogarit shumën e disa vlerave në qelizat e njëpasnjëshme!



# 1.7 Sortimi i të dhënave

Kur fleta e punës përmban shumë përmbajtje, është vështirë që të gjenden shpejt informacionet. Prandaj, mund të përdoren filtrat për të ngushtuar (reduktuar) të dhënat në fletën e punës, duke lejuar që të shihen vetëm informacionet e nevojshme nga përdoruesi.

Të dhënat në tabelë mund të organizohen dhe pastaj të krijohen raportet në përputhje me kërkesën e përdoruesit. Kjo realizohet me operacionet si në vijim:

- sortimi i të dhënave në një radhitje të caktuar;
- filtrimin e të dhënave sipas kritereve të caktuara;
- krijimin e të dhënave të përmbledhura dhe në nënbashkësi.

Në këtë njësi mësimore do të përqëndrohemi në procedurën e sortimit, si mjet i veçantë që lehtëson qasjen në informacionet e nevojshme si të ndara nga të tjerat.

Sortimi, nënkupton ndryshimin e radhitjes së rreshtave në tabelë, në mënyrë që të dhënat të jenë në radhitje të caktuar. Në Excel janë të mundshme sortimet si në vijim:

- sortimi në radhitje rrënëse ose rritëse;
- sortimi alfabetik, numerik, kronologjik ose sipas radhitjes së përcaktuar nga përdoruesi;
- sortimi sipas definimit për formatimin me kusht.

Nëse sortimi duhet të bëhet sipas një kriteri, përkatësisht sipas një kolone, atëherë mjafton të klikohet vetëm në fushën e kolonës sipas të cilës përdoruesi dëshiron të sorton dhe pastaj nga menyja Home zgjedhim Sort & Filter.

Inform	natika Informatika	₽, s	
∑ AutoSum → ↓ Fill → ≪ Clear →	Sort & Find & Filter * Select *		
Editi	<u>A</u> ↓ Sort A to Z	1	Sortimi sipas radhitjes alfabetike
	Z↓ Sort Z to A	2	Sortimi i anasjelltë sipas radhitjes alfabetike
	Custom Sort.	3	Sortimi sipas kushtit të caktuar
Q K	Filter		
	📡 <u>C</u> lear		
	🌄 Reapply		

Figura 1: Sortimi

Nëse përdoruesi zgjedh opsionin **Custom Sort**, atëherë në fushën **Column (Sort by)** zgjedhet kriteri sipas të cilit do të kryhet radhitja, në fushën **Sort on** zgjedhet tipi i të dhënave që do të sortohen, dhe në fushën 3 zgjedhet radhitja e sortimit. Për shembull, në tabelën e mëposhtme është bërë sortimi sipas kritereve në vijim:

1. sipas mbiemrit të nxënësit sipas rendit alfabetik;

2. suksesi mesatar si kriteri i dytë sipas radhës, nga vlera mesatare më e madhe kah më e vogla.



Figura 2: Sortimi i të dhënave me Custom Sort

# Detyrë

Në programin për llogaritje tabelare të krijohet tabela si në vijim "Të dhënat e bibliotekës":

a Të huazuara 84
a Të huazuara 84
84
68
35
44
54
30
57
67
46
44
70
-

Të kryhet sortimi sipas kategorisë së "**Lektyrës**" sipas rendit alfabetik, dhe pastaj sa herë është "**Huazuar**" duke filluar nga vlera më e madhe kah më e vogla!

	А	В	B C D		E	F
1	10 10 10 10 10 10 10 10 10 10 10 10 10 1	Të c	lhëna për bibli	otekën	1998 - 19	
2	Nr.rendor	Lektyra	Emri i autorit	Mbiemri i autorit	Klasa	Të huazuara
3	1	Nëna	Jovan	Jovanoviq-Zmaj	II	84
4	2	Zoki Poki	Olivera	Nikolovska	II	68
5	3	Poezia	Rifat	Kukaj	III	35
6	4	Tregim i sheqeri	Slavko	Janeski	III	44
7	5	Tregimi për fëmijën Vilen	Gligor	Prliçev	IV	54
8	6	Shkolla	Dragan	Lukiq	III	30
9	7	Orhan	Nexhati	Zekirija	IV	57
10	8	Lulet e verës	Naim	Frashëri	VI	67
11	9	Siljan Lejleku	Marko	Cepenkov	VII	46
12	10	Princeza Argjiro	Ismail	Kadare	VII	44
13	11	E ikura	Video	Podgorec	VIII	70



# Mbaj mend!

Sortimi është proces i klasifikimit të të dhënave sipas rregullave. Radhitja mund të bëhet në rend alfabetik, në rend alfabetik të anasjelltë, ose sipas kushteve të definuara të cilat jepen përmes opsionit **Custom Sort**.



- 1. Çfarë është sortimi?
- 2. Čila është procedura për radhitjen e të dhënave sipas rendit alfabetik?
- 3. Si mund t'i radhitim të dhënat me një kusht të caktuar?

# 1.8 Krijimi i grafikut

Pasi tabela të jetë krijuar, dhe me format sipas procedurave që i kemi theksuar në njësitë mësimore të mëparshme, shpeshherë është e nevojshme që të dhënat të paraqiten me figura, përkatësisht duke përdorur grafikun. Me paraqitjen e të dhënave me figurë i mundësohet përdoruesit që të shohë shpejt dhe qartë vlerat e të dhënave, krahasimin dhe analizën e tyre.

Sipas kësaj, **grafikët janë paraqitje me figura (imazhe) e të dhënave nga tabela**. Në fakt, me grafikun të dhënat shndërrohen në figurë (Imazh). Grafikët mund të jenë: me formë shtylle, në formën e vijave, grafiku në formë tortë etj., pra, grafiku na nevojitet për qëllime të ndryshme, si për shembull: për krahasim sipas kategorive, për krahasim të vlerave të përqindjes etj, siç tregohet në figurën vijuese:



Figura 1: Llojet e grafikut

Procedura për krijimin e grafikut në programin për llogaritje tabelare është e thjeshtë, nëse të dhënat janë të shkruara në formë të rregullt dhe janë të organizuara në fletën e punës. Fillimisht, selektohet rangu i qelizave me të dhënat të cilat duhet të paraqiten me grafik. Pas zgjedhjes së të dhënave, të cilat do të paraqiten me grafik mund tëshfrytëzojmë urdhërin për krijimin e grafikut me **Excel 2016**, përmes menysë **Insert** dhe **Charts**:

Ŧ							Book1 -	Excel	
1 Insert	Page Layout	Formulas	Data	Review	View	♀ Tell me what	you want to	o do	
ended Table	Pictures Onli Pictu	ne irres 💁 T	🗎 Store 🕽 My Add-	ins + Red	commended Charts	uli • III • rši ∭ • uli • iši ● • ⊡ • ★	PivotChar	rt 3D Map ▼	1 L
	Illustrati	ons	Add-in:	s		Charts 2		Tours	
: × .	f_r							See All Cha	arts

Figura 2: Procedura e paraqitjes së grafikut

Nga figura e shohim vendosjen e grafikut, ku menjëherë mund të zgjedhim edhe kategorinë, përkatësisht llojin e grafikut, ose përmes **Quick Launcher**, që gjendet në anën e majtë të rubrikës **Chart**, dhe pastajshfaqet dritarja si në vijim:



Figura 3: Procedura për zgjedhjen e llojeve të grafikëve

Nëse klikoni në butonin **OK**, grafiku paraqitet në fletën e punës, ku gjendet burimi i të dhënave, përkatësisht tabela burimore. Atëherë marrim vendim, nëse grafiku duhet të mbetet në të njëjtën fletë, ose ta bartim në një fletë të re. Kur grafiku është i selektuar, atëherë paraqiten menytë **Design** dhe **Format**, të cilat mundësojnë formatimin dhe rregullimin (editimin) e pamjes së grafikut. Për shembull, përmes menysë **Design** mund të aplikohen mundësitë si në vijim:

- shtimi i elementeve në grafik;
- radhitja e elementeve në grafik;
- zgjedhja e ngjyrës;
- stili i grafikut;
- zgjedhja e të dhënave;
- ndryshimi i llojit të grafikut;
- bartjen e grafikut në një fletë të re të punës/në flet të njëjtë të tabelës.

Përmes menysë **Format** mund të shtohen forma të pjesëve të grafikut, ngjyra e vijës, ngjyrosja e elementeve, stilet e shkronjave, dimensionet e elementeve etj.

# Shënim!

Gjatë krijimit të formulave, nuk do të punojmë me vlerat e qelizave, por me adresat e tyre, me qëllim që me ndryshimin e vlerave të të dhënave në mënyrë automatike të ndryshohet edhe rezultati.



#### Detyrë

Le të hapim tabelën "Të dhënat nga biblioteka" që ruhet në folderin e sipërfaqes së punës. Të krijojmë grafikun, kudo të paraqitet pamja se cili libër ka qenë më shumë i lexuar, për këtë rast le të përdoret grafiku Columns, siç është paraqitur në figurën vijuese:



Grafiku le të jetë në fletë të re dhe të formatohet duke aplikuar mjetet nga menyja **Design** dhe **Format**.



Grafiqet janë paraqitje me figura (imazhe) për të dhënat në tabelë. Ato mundësojnë pamjen e qartë të vlerave të të dhënave, krahasimin e tyre, etj.



- 1. Çfarë është grafiku?
- 2. Përshkruani procedurën për krijimin e një grafiku?
- 3. Cilat lloje të grafikëve ofron programi Excel 2016?
- 4. Cilat mjete mund të përdoren nga menyja **Design** dhe **Format** për të rregulluar dhe formatuar grafikun?

# TË PËRSËRISIM! TË USHTROJMË



# Detyra 1

Hapni një fletë pune në MS Excel 2016. Emërtoni fletën e parë të punës "shtëpi" dhe në të, duke përdorur mënyra të ndryshme të përzgjedhjes, ngjyrosni qelizat ashtu që të fitoni shtëpi. Dokumentin e ruani me mbiemrin tuaj dhe klasën!



# Detyra 2

Në programin për llogaritje tabelare në fletën e punës 1 (Sheet 1) krijoni një listë për nxënësit e klasës dhe numrat e telefonave të tyre. Tabela duhet të përmbajë të dhënat si në vijim: numri rendor, emri, mbiemri dhe telefoni. Pas (krahas) telefonit të shokut/shoqes tuaj, shtoni edhe ditëlindjen e tyre.



# Detyra 3

Bëni një listë të librave në bibliotekën tuaj. Vendosni të dhënat vijuese: Autori, titulli i librit dhe viti i botimit



# Detyra 4

Krijoni një tabelë në të cilën do të futen të dhënat: muaji, të ardhurat, shpenzimet. Pas vendosjes së të dhënave shtoni një kolonë të re: ndryshimi (ndryshimi në mes të ardhurave dhe shpenzimeve). Mbi qelizat me tituj të thekuar mësipër, shtoni një rresht të ri. Ngjyrosi qelizat e rreshtit të ri me Fill color. Rreshtave u jepni dimensionet njëzet (20), ndërsa kolonave me dimensionet njëzet e pesë (25). Ruani dokumentin!



# Detyra 5

Në dokumentin e punës të programit për llogaritje tabelare krijoni tabelën si në vijim:

E	5 🔹 !	$\times \checkmark f_s$	¢		
	A	В	с	D	E
1		Sukse	esi sipas lëndë	ive	
2	Lënda	Të notuar	Të pa notuar	Nota pozitive	Nota negative
3	Matematikë	12	1	10	2
4	Informatikë	11	2	11	0
5	Gjuhë angleze	10	3	10	0
6	Gjuhë amëtare				

Formatoni tabelën duke përdorur mjetet për formatim: fonti, madhësia, stilet, rregullimi, orientimi, si dhe zbatimi i stileve në qelizat ose formatet e gatshme në tabelë. Dokumentin e ruani!



#### Detyra 6

Në programin për llogaritje tabelare të krijohet tabela si në vijim me zbatim të fomrulave për llogaritje të perimetrit dhe sipërfaqes:

A	2	× v	f <sub>x</sub>		
	А	В	С	D	E
1	F	igurat gje	ometrike	2	
2		Brinja 1	Brinja 2	Perimetri	Sipërfaqja
3	Drejtëkëndëshi	6	4		
4	Njëra faqe e kubit	3	3		
5	Katrori	5	5		



# Detyra 7

Të krijohet programi për llogaritje tabelare ku janë paraqitur temperaturat mesatare të shteteve, sipas muajëve. Me ndihmën e funksioneve përcaktoni cili muaj është më i nxehtë, dhe cili muaj është më i ftohtë gjatë vitit!

F5		*	:	×		
	A					
1	Janar			5.6		
2	Shkurt		1	6.2		
3	Mars			8.8		
4	Prill	12				
5	Maj	17				
6	Qershor			21		
7	Korrik			23		
8	Gusht			23		
9	Shtator			18		
10	Tetor			15		
11	Nëntor			10		
12	Dhjetor		3	6.9		
	S	12				

# Detyra 8

Krijoni një tabelë me të dhëna për shokët tuaj: emri, mbiemri, mosha dhe sortoni ato sipas renditjes alfabetike të anasjelltë (të kundërt).



#### Detyra 9

Në programin për llogarijte tabelare krijoni dhe formatoni tabelën me të dhëna për shokët tuaj: emri, mbiemri, suksesi. Sortoni sipas mbiemrit, sipas rendit alfabetik, dhe më pas, sipas suksesit, duke filluar nga më i madhi te më i vogli.



# Detyra 9

Në programin për llogarijte tabelare krijoni dhe formatoni tabelën me të dhëna për shokët tuaj: emri, mbiemri, suksesi. Sortoni sipas mbiemrit, sipas rendit alfabetik, dhe më pas, sipas suksesit, duke filluar nga më i madhi te më i vogli.



# Detyra 10

Tëkrijohetgrafikunëprograminpërllogaritjetabelare, kudotëkrahasohetsuksesi i klasave nga VI - IX. Pastaj, të aplikohet grafiku tortë (pie chart), në një fletë të re.



# Detyra 11

Krijoninjë tabelë që tregon suksesin mesatar në fund të çdo tremujori, si çtregohet në figurë. Pastaj, paraqitni grafikisht të dhë nat duke përdorur një grafik Columns.





#### Detyra 12

Hulumtoni sa është çmimi i ujit për metër katror në qytetin tuaj dhe njehësoni çfarë pagese duhet të paguani.

# Në programin për llogaritje tabelare, në fletën e punës të krijohet tabela si në vijim:

Nr.rendor	Mbiemri	Emri	Qyteti	Gjuhë shqipe	Gjuhë maqedone	Gjuhë angleze	Gjuhë gjermane
1	Kadriu	Lulart	Tetovë	5	4	5	4
2 Atanasoska Ankica Shkup		Shkup	5	5	5	5	
3	Alili	Hekuran	Gostivar	3	4	4	3
4	llievski	Ilija	Manastir	2	3	5	4
5	Osmani	Baki	Shkup	4	4	4	5
6	6 Milovska Irena		Shtip	5	3	5	5
7	Nestoroska	Ana	Gostivar	3	4	4	4
8	Hiseni	Afërdita	Veles	5	3	5	3
9	Pejovski	Ana	Manastir	5	5	3	4
10	Saiti	Nazire	Shkup	4	4	4	5
11	Simoski	Riste	Gostivar	3	3	5	4
12	Etemi	Sara	Shkup	4	4	4	3
13	Asani	Drita	Dibër	5	3	5	4
14	Spiroski	Riste	Valandovë	4	4	4	5

- Titujt e qelizave të gjuhëve të kenë drejtim vertikal të tekstit dhe të zbatohet opsioni Wrap Text.
- Menjëherë pas kolonës "Gjuhë gjermane" të shtohet kolona "Nota më e madhe"" dhe me ndihmën e funksionit të njehësohet nota më e madhe.
- Pas kolonës "Nota më e madhe" të shtohet kolona "Nota më e vogël" dhe me ndihmën e funksionit të njehësohet nota më e vogël.
- Në fund të tabelës të llogaritet nota mesatare për çdo nxënës/student.
- Qelizat A15 deri J15 të bashkohen dhe të shkruhet mesatarja e klasës.
- Nxënësit me mesataren "mirë" të ngjyrosen me ngjyrë të kuqe, ndërsa nxënësit me mesatare "shkëlqyeshëm" të ngjyrosen me ngjyrë të gjelbër, duke shfrytëzuar formatimin.
- Në qelizën K15 të gjendet mesatarja e klasës.
- Nën kolonat e lëndëve të numërohen sa nota ka me: pesë, katër, tresh dhe sa dysha për çdo lëndë në veçanti.
- Tabela të formatohet duke përdorur: font, ngjyrë të shkronjave, stile të shkronjave, ngjyrë të prapavijës (background) së qelizave, lloji i kornizës dhe ngjyra e kornizës me kusht që të tregoj dukshmëri. Mund të përdoren dhe formate të gatshme/dizajn të tabelës.
- Sortoni të dhënat sipas mbiemrit duke filluar nga A, ndërsa kushti i dytë sipas suksesit duke filluar nga më i miri.
- Krijoni grafikun në fletën e punës të veçantë, ku do të paraqitet suksesi i nxënësit. Grafikun e formatoni sipas dëshirës.
- Ruani dokumentin!

# Njohja me konceptet informatike duke zgjidhur detyra logjike konkurruese

Njohuri për konceptet informatike duke zgjidhur detyra logjike konkurruese

Analiza dhe zgjidhja e detyrave logjike konkurruese

Lidhja e detyrës me konceptet nga shkenca kompjuterike - konceptet e informacionit

- Strukturat e të dhënave
- Numrat binarë
- Kriptografia

#### TË PËRSËRISIM! TË USHTROJMË!

# 2 Njohja me konceptet informatike duke zgjidhur detyra logjike konkurruese

### Çfarë do të mësojmë?

Të krijojmë tabela me tipe të ndryshme të të dhënave dhe një seri të llogaritjeve duke përdorur formula dhe funksione, të rregulluara me mjete për formatim dhe të paraqitura grafikisht nga grafikët.

Kur studioni shkenca kompjuterike, ndër të tjera, një nga gjërat më emocionuese dhe interesante është të mësosh një mënyrë të re të të menduarit dhe zgjidhjen e detyrave dhe situatave problemore, dhe kjo quhet të **menduarit kompjuterik**. Kjo mënyrë e të menduarit është thelbësore për shekullin e 21-të, si bazë për arritjen e suksesit në shumë situata në jetën e përditshme, gjatë arsimimit, si dhe në zhvillimin dhe ngritjen profesionale.

Para se të përdorni kompjuterët për të zgjidhur një problem ose situatë të ndonjë problemi, është e nevojshme të kuptoni thelbin e problemit. Prandaj, mund të thuhet se të menduarit kompjuterik është një përmbledhje e disa aftësive, të tilla si: kreativiteti, mundësia për shpjeguar dhe vërtetësia, puna në ekip dhe bashkëpunimi.

Elementet kryesore e të menduarit kompjuterik janë:

- të menduarit logjik;
- të menduarit algoritmik;
- zgjidhje efikase;
- të menduarit shkencor;
- të menduarit inovativ.

Në këtë temë do t'i kushtojmë një theks të veçantë të **menduarit logjik**, si një aftësi e veçantë dhe themelore e të menduarit kompjuterik. Mendimi logjik shpesh shoqërohet me konceptet kompjuterike, por së fundmi, është bërë sinonim i integrimit të ideve kompjuterike në disiplina të tjera.

Kur themi se diçka është logjike, në të vërtetë mendojmë se kjo gjë ka kuptim. Aftësia e një individi për të menduar të bazuar në fakte dhe prova njihet si aftësia logjike për të menduar. Prandaj, **të menduarit logjik mund të përkufizohet si një proces që përdor arsyetimin objektiv në mënyrë që të arrihet deri në përfundim të caktuar**.



# Mundohuni!

Hulumtoni termin logjika! Definoni atë dhe përpiquni të hulumtoni ndryshimet ndërmjet të menduarit logjik dhe mendimit kritik? Çfarë është e ngjashme dhe çfarë është e ndryshme midis tyre.

Detyrat ose situatat problemore që zgjidhen me ndihmën e të menduarit logjik kanë një strukturë, lidhje midis fakteve dhe kanë një kuptim, pra një marrëdhënie logjike shkak-pasojë. Kjo mënyrë e të menduarit përfshinë **analizën e thelluar**, si për shembull: vlerësimi i të gjitha opsioneve në dispozicion, përdorimi i fakteve dhe numrave si dhe marrja e vendimeve të rëndësishme të bazuara në anët e mira dhe jo të mira, që do të thotë, se gjatë këtij procesi nuk merren parasysh elementet e ndjenjave dhe emocioneve. Shembulli më i zakonshëm i të menduarit logjik është **SUDOKU**. Janë dhënë informacione rreth vlerave që përmbahen në disa qeliza. Për të zgjidhur enigmën duhet të mbyllen vlerat në të gjitha qelizat tjera, duke respektuar rregullat gjatë zgjidhjes së problemit të detyrës. Nëse bëhet gabim më i vogël, atëherë do të jemi larg zgjidhjes.

Edhe pse ky proces është i padukshëm, të gjithë ne përballemi me situata dhe gjendje të ndryshme që çdo ditë i tejkalojmë falë aftësive tona të arsyetimit. **Për shembull**: ndërsa llogaritim çmimet në supermarket për të parë nëse mund të marrim gjithçka që na nevojitet për një çmim më të ulët, ose ndërsa përpiqemi të vendosim të gjitha përgjegjësitë tona në një ditë, "makina jonë e të menduarit" rrotullohet vazhdimisht për të gjetur zgjidhjen e duhur. Prandaj, në të menduarit logjik, gjërat më të rëndësishme që duhet të patjetër t'i respektojmë janë si në vijim:

- Mos i shikoni gjërat vetëm nga perspektiva juaj. Është subjektive dhe nuk na çon te qëllimi, pra te zgjidhja përfundimtare. Për shembull: Stefani dhe Almiri drekojnë në një restorant. Ena e Stefanit ka një erë të pakëndshme, ndërsa Almiri e shijon drekën. Meqenëse Stefanit nuk i pëlqente aroma e ushqimit, ai shpejt arriti në përfundimin se ushqimi nuk është i shëndetshëm dhe nuk është përgatitur mirë. Kjo nuk është mënyra logjike për të arritur në një përfundim, sepse Stefani nuk ka asnjë provë se ushqimi është jo i shëndetshëm dhe se nuk është përgatitur mirë. Për të arritur në përfundim logjik, njeriu duhet të përjashtojë mendimet e veta subjektive dhe të përqendrohet në informacione të provuara, siç janë përgatitjes dhe të mos mbështetet në supozimet.
- Mendoni para se të filloni Krijoni një strategji. Strategjia luan një rol të madh në procesin e të menduarit. Filloni kërkimin tuaj duke bërë pyetje që do t'ju ndihmojnë të interpretoni faktet. Kërkoni në mënyrë aktive për detaje dhe mësoni se si ato funksionojnë si pjesë të ndara, dhe si një grup, para se të arrini deri te figura (imazhi) e madhe.

 Kushtoni vëmendje kuptimit të fjalëve. Ndryshimet e vogla gjuhësore bëjnë ndryshim të madh në kuptimin e detyrës ose situatës problemore. Përcaktimi i ndryshimit midis deklarimeve do të eleminojë patjetër mospërputhjet dhe paqartësitë në të menduarin logjik. Për shembull: fjala "e domosdoshme" është e ndryshme nga "e mjaftueshme". E domosdoshme do të thotë që kushti ose procedura duhet të plotësohen, në krahasim me të mjaftueshme, që do të thotë se duhet bërë një përpjekje minimale për të arritur një zgjidhje.

Në fund, të konkludojmë se të menduarit logjik si një grumbull i aftësive dhe teknikave për zgjidhjen e detyrave dhe situatave me probleme, është një hap para programimit si një proces i krijimit të softuerit/programeve për qëllime të ndryshme.



#### Termat kryesorë!

librat e punës, fleta e punës, tabela, kolona, rreshti, qeliza, qelizat e njëpasnjëshme/ jo të njëpasnjëshme, qeliza aktive, formula, funksioni, grafiku.



#### Mbaj mend!

Të menduarit kompjuterik është një mënyrë e re për zgjidhjen e situatave dhe detyrave problemore, të cilat me ndihmën e të menduarit logjik arrijmë në një zgjidhje objektive. Të menduarit logjik bazohet në fakte dhe prova deri te të cilat arrihet në mënyra të ndryshme, duke përdorur teknika dhe metoda të ndryshme. Kjo mënyrë e të menduarit kurrë nuk bazohet në supozime, dëshira, ndjenja dhe gjendje emocionale.



# 2.1 Analiza dhe zgjidhja e detyrave logjike konkurruese

**Analiza** e problemit ose detyrës logjike është hapi i parë themelor që duhet në mënyrë objektive ta bëjmë, që të fillohet me rrugën drejt zgjidhjes përfundimtare. Me ndihmën e **analizës** zbërthejmë detyrën ose situatën problemore, zbulojmë fakte ose elemente të tërësisë, zbulojmë lidhshmëritë me qëllim që të vijmë te rezultati.

Analiza e detyrave përfshinë tre hapa që mund të paraqiten si në vijim:



Figura 1: Hapat gjatë analizës së detyrave problemore

Nëse analiza e detyrave është e organizuar me kujdes dhe ndjek hapat e duhur, kjo mund të zbatohet për të zgjidhur shumë detyra dhe për të arritur zgjidhjen përfundimtare, pikërisht për shkak të analizës së detajuar, të kujdesshme dhe të integruar.



# Mundohuni!

Mundohuni të hulumtoni se cila është e anasjelltë ose proces i kundërt i analizës, nëse e dimë që analiza është një proces i ndarjes së detyrës në elemente të veçantë ose në pjesë!

Le të analizojmë disa detyra problemore, duke zbatuar disa teknikat që i kemi theksuar, pastaj zbatojmë edhe njohuritë e fituara në lidhje me të menduarit logjik.



# Detyrë projektuese

Jovani, Astreti dhe Samiu janë bashkëmoshatarë dhe vijojnë të njëjtën shkollë. Të tre janë atletë të mirë që kanë fituar shumë çmime në gara kombëtare dhe ndërkombëtare. Cili prej tyre është më i ngadalshëm, nëse të gjitha deklarimet e theksuara janë të sakta?

- 1. Jovani nuk është më i ngadalshmi;
- 2. Astreti është më i shpejti;
- 3. Samiu nuk është më i shpejti.



Në një nga deklaratat thuhet se **Astriti** është më i shpejti dhe për këtë arsye, ne e lidhim karakterin e **Astritit** me katrorin e shënuar si "Më i shpejti". Tani janë dy fusha të mbetura zbrazët. Gjithashtu thuhet se **Jovani** "nuk është më i ngadalshmi" dhe kjo është arsyeja pse e lidhim atë me katrorin e dytë. Vetëm fusha e parë e shënuar "Më i ngadalshmi" mbetet e zbrazët dhe e lidhim atë vetë. Prandaj, në pyetjen se kush është më i ngadalshëm, përgjigjja e saktë është **Samiu**.

Në zgjidhjen e detyrës së caktuar kemi përdorur teknikën e **abstraksionit**, sepse kemi nxjerrë karakteristikat më të rëndësishme të detyrës. Karakteristikat e identifikuara na udhëzuan se nga ku të fillojmë të shqyrtojmë problemin e detyrës, dhe cilën rrugë duhet të ndjekim që të arrijmë në zgjidhjen e mundshme. E njëjta gjë ndodh gjatë zgjidhjes së një situate problemore në **informatikë**. Gjithmonë bëhet fjalë për informacionet që janë të rëndësishme për zgjidhjen e situatës problemore dhe informacioni i panevojshëm largohet ose neglizhohet gjatë zgjidhjes së problemit.



# Detyrë

Ngjashëm me detyrën e mëparshme, përpiquni të gjeni vetë zgjidhjen e saktë të detyrës si në vijim! Nëse të gjitha pohimet janë të vërteta, ku është pasuria?



Pasuria nuk është tek 2





Pasuria është tek 1 ose 3

Pasuria nuk është këtu



Sudoku është një nga lojërat më të njohura që përfshinë të menduarit logjik dhe procesin e eliminimit të fakteve që nuk janë në interes për të arritur deri te rezultati.

Le të zgjidhim një detyrë të thjeshtë Sudoku!



Në tabelë, në disa fusha janë radhitur figurat gjeometrike, dhe disa fusha janë të zbrazëta. Fushat e zbrazëta duhet të plotësohen me figurat si në vijim:



Çdo figurë gjeometrike duhet të ketë të njëjtin numër figurash, të radhitura në atë mënyrë që horizontalisht dhe vertikalisht të ketë gjithmonë vetëm një figurë e të njëjtit lloj. Vërejmë se rrethi mund të shtohet në tabelë vetëm një herë dhe prandaj duhet të përcaktohet vendi i duhur, përkatësisht në drejtimin horizontal dhe vertikal duhet të ketë vetëm një figurë të tillë. Le të gjejmë vendin e duhur. Në rreshtin e parë, të tretë dhe të katërt (horizontalisht), ekziston tashmë figura rrethi, kështu që si zgjedhje mbetet rreshti i dytë.

#### Por në cilën fushë në rreshtin e dytë? Në cilën kolonë?

Kolonat një, dy dhe tre tashmë përmbajnë figurën rrethi dhe ato përjashtohen nga opsioni. Prandaj, si një mundësi është të hjekim rreshtin e parë, të tretë dhe të katërt, si dhe kolonën e parë, të dytë dhe të tretë. E vetmja mundësi mbetet të shtojmë figurën e rrethit në rreshtin e dytë dhe në kolonën e katërt, dhe kjo është përgjigjja e saktë. Nëse plotësojmë të gjitha fushat, sipas shembullit me rrethin, atëherë fitohet rezultati vijues:





# Mundohuni!

Mundohuni të zgjidhni Sudoku duke zbatuar rregullat për plotësimin e fushave me numra! Ka shumë shembuj elektronikë, por në mediumet e shtypura mund të gjenden në versionin e shtypur.

Procesi i **eliminimit**, gjatë zgjidhjes së detyrave logjike konkurruese, lejon përjashtimin e fakteve që nuk janë domethënëse dhe që nuk na çojnë në zgjidhjen përfundimtare të situatës problemore, si në rastin Sudoku. Në jetën e përditshme, eliminimi përdoret më së shpeshti në zgjidhjen e testeve që ofrojnë përgjigje të shumta të mundshme, duke eleminuar kështu përgjigjet e pasakta për të zvogëluar numrin e opsioneve të mundshme për përgjigje të saktë.

Prandaj, gjatë analizës së detyrave logjike, shfrytëzuam teknikat e të menduarit logjik për të arritur një zgjidhje përfundimtare. Këtu janë shfrytëzuar teknikat e abstraksionit, eliminimit dhe respektimit të rregullave të dhëna. Sidoqoftë, gjatë zgjidhjes së situatave dhe detyrave problemore, si pjesë informatike ose proceseve kompjuterike, përdoren teknika të tjera që rrjedhin nga nevoja dhe natyra e detyrës, me qëllimin e vetëm, që në mënyrën më optimale, të arrihet në një zgjidhje përfundimtare.

# Mbaj mend!

Analiza e detyrave dhe situatave problemore duhet të jetë objektive dhe jo të njëanshme, që do të thotë nuk duhet të përfshihen elementet e subjektivitetit. Procedura e analizës përfshinë tre hapa: zbërthimin e detyrës në elemente përbërëse, përcaktimin e marrëdhënieve midis elementeve dhe ristrukturimin sipas rezultateve të fituara. Teknikat më të zakonshme në analizimin e një detyre ose situate problemore janë: abstragimi, eliminimi, respektimi i rregullave, gjetja e ngjashmërive, etj.

# 2.2 Lidhja e detyrës me konceptet nga shkenca kompjuterikekonceptet informatike

**Të menduarit logjik** dhe zgjidhja e detyrave logjike konkurruese janë aftësi që mund të përdoren në shumë disiplina shkencore, por më shumë lidhen me kompjuterët dhe **konceptet informatike**. Prandaj, nuk është rastësi që të menduarit logjik është një hap para **programimit**, si një proces i krijimit të zgjidhjeve aplikative.

Prandaj, **programimi përcaktohet si procesi i shkruarjes së një programi duke përdorur gjuhët programuese**. Personat që krijojnë programe quhen **programues**.

Që nga fillimi i krijimit të programit, programuesit kryejnë hapa të shumtë, ku secili në një mënyrë të veçantë i hapin rrugën zgjidhjes, pra krijimit të programit. Në këtë proces, programuesit njihen me konceptet e informacionit. Prandaj, në këtë përmbajtje mësimore do të njihemi me disa prej tyre.

#### 2.2.1 Struktura e të dhënave

Strukturat e të dhënave janë formë e veçantë ose mënyrë për mbledhjen, organizimin, përpunimin dhe ruajtjen e të dhënave. Strukturat më të përdorura janë: vargjet, listat e lidhura, steku/ grumbujt, rreshtat, rreshtat prioritar, pema binare, etj.

Sipas kësaj, strukturat e të dhënave mund të përdoren për të krijuar:

- Lista me hapa për të ardhur deri te zgjidhja e caktuar.
- Varg instruksionesh për të arritur deri te ndonjë destinacion.
- Udhëzim për të plotësuar ndonjë formular ose skemë, etj.



# Detyrë

Të krijohet vargu me hapa, duke përdorur shenjat e theksuara me qëllim të fitohet figura si në vijim:





Hapi 1	2	3	4	5	6	7	8	9	10
Hapi 2	12	13	14	15	16	17	18	19	20
Нарі З	22	23	24	25	26	27	28	29	30

Përmes detyrës së zgjidhur, kemi arritur të krijojmë një seri udhëzimesh për zgjidhjen përfundimtare, që do të thotë ngjyrosjen e rrjetit të katrorëve sipas figurës së paraqitur.



# Detyrë

Të krijohet vargu me hapa, duke përdorur shenjat e theksuara me qëllim të fitohet figura si në vijim.



Qëllimi është që të ndërmerren hapa për të lëvizur disqet nga një vend në tjetrin, pastaj të kujdesemi per radhitjen.

Hapi 1: Diskun me numër 1 e transferojmë në rreshtin e tretë.

Hapi 2: Diskun me numër 2 e transferojmë në rreshtin e dytë.

Hapi 3: Diskun me numër 1 e transferojmë në rreshtin e dytë, menjëherë mbi diskun me numrin 2.

Hapi 4: Diskun me numër 3 e transferojmë në rreshtin e tretë.

Hapi 5: Diskun me numër 1 e transferojmë në rreshtin e parë.

Hapi 6: Diskun me numër 2 e transferojmë në rreshtin e tretë, mbi diskun me numër 3.

Hapi 7: Diskun me numër 1 e transferojmë në rreshtin e tretë, mbi diskun numër 2.

Sipas hapave të deklaruar, arritëm të zhvendosim të gjithë disqet nga një vend në tjetrin, duke i kushtuar vëmendje rregullave të lëvizjes të parashikuara nga loja. Me shtatë hapa, i zhvendosëm disqet nga rreshti i parë në rreshtin e fundit, dhe disqet mbetën me radhitje të njëjtë.



Gjatë kryerjes së detyrës "Kullat në Hanoi", u njohëm me një strukturë stek/ grumbull të dhënash, ku karakteristikë kryesore është rregullimi, ashtu që fillimisht merren elementet që janë në kulm. Në fakt, për "stekun" themi se respektohet procedura "i pari hynë, i fundit del". Sigurisht, gjatë transferimit ndjek rregulla që vijnë nga vetë natyra e detyrës.



#### Mundohuni!

Me ndihmën e një mësuesi të biologjisë, përpiquni të krijoni një varg në të cilën do të tregoni ciklin jetësor të një fluture. Sa periudha ka cikli jetësor i një fluture? A mund të anashkalohet ndonjë prej këtyre cikleve.

#### 2.2.2 Numrat binarë



#### Të rikujtojmë

A ju kujtohet se çfarë është përpunimi i të dhënave kompjuterike? Te të dhënat cilat janë përparësitë e përpunimit kompjuterik? A mund ta shpjegoni termin përpunim "automatik"?

Sistemi i numrave binarë përbëhet nga dy vlera numerike, shifrat 0 dhe 1. Vetëm një shifër 0 ose 1 quhet **bit**. Me një bit, mund të paraqesim gjendjen e "saktë" ose "të pasaktë", "po" ose "jo", "të zënë" ose "të lirë" dhe shumë situata të tjera.

Me ndihmën e numrave binarë, të dhënat në kompjuter paraqiten në **vargje të zerove dhe njësheve**. Shndërrimi i shkronjave në një fjalë në numra binarë paraqet **kodin binar**. Çdo numër natyror mund të përfaqësohet nga një seri numrash binarë, për shembull:

Numrat natyror	0	1	2	3	4	5	6	7	8	9	10
Paraqitja binare	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010

Për të shndërruar (konvertuar) një numër natyror në një paraqitje binare, atëherë përdorim bazën 2 me shkallë: 0, 1, 2, 3, etj. Për shembull, numri 5 në binar do të dukej kështu: 0101, dhe ky rezultat vjen nga:

	Chkollo mo hozë 2	2³ = 8	2² = 4	2 <sup>1</sup> = 2	2º = 1
5	SIIKalla IIIe Daze z		4 + 1	1 = 5	
	Paraqitja me bita	0	1	0	1

Le të plotësojmë enigmën e mëposhtme në mënyrë që në secilën fushë të shkruajmë përfaqësimin binar të numrit natyror:



Me ndihmën e të menduarit logjik ta zgjedhim detyrën konkurruese si në vijim:



#### Detyrë

Nxënësit e klasës VII kanë zgjidhur testin nga gjeografia që përmbante pyetje me shumë përgjigje. Mësuesi regjistroi përgjigjet e sakta me 1 dhe ato të pasakta me 0 në tabelë, siç tregohet më poshtë.

	Pyetja 1	Pyetja 2	Pyetja 3	Pyetja 4	Pyetja 5	Pyetja 6
Ana	0	0	1	1	1	0
Sara	1	1	1	1	1	1
Almir	1	0	0	0	1	1
Safet	0	0	0	0	1	1
Elona	0	0	1	1	0	1
Amira	1	0	1	0	1	1
Jelena	0	0	1	1	1	1

Nga tabela, mësuesi dëshironte të krijonte një raport për atë se cili material është më i dobët dhe cili është më shumë i përvetësuar, në mënyrë që të krijoj mënyra shtesë për përvetësimin e plotë të materialit.

Përgjigjuni pyetjeve sipas të dhënave që janë theksuar në tabelë!

- Në cilën pyetje u përgjigjën më shumë nxënës?
- Në cilën pyetje u përgjigjën më pak nxënës?
- Cili nxënës ka rezultatin më të lartë?
- Kush është nxënësi më i dobët?
- Cilët nxënës duhet të ndjekin mësimin shtesë sipas rezultateve?

### 2.2.3 Kriptografia

Kriptografia është një disiplinë shkencore që merret me studimin e metodave për dërgimin dhe pranimin e mesazheve në mënyrën që është e kuptueshme vetëm për ata për të cilët është dedikuar. Në fakt, kriptografia mundëson përdorim të kodeve sekrete, me çka përdoruesve u garanton:

- sigurinë e të dhënave nga qasja e paautorizuar;
- komunikimin e sigurtë dhe të mbrojtur;
- besueshmërinë dhe fshehtësinë e të dhënave;
- verifikimin dhe prejardhjen e të dhënave, përkatësisht informacioneve etj.

Kriptografia përdoret gjatë punës me kartela kreditore, fjalëkalime të kompjuterit, tregti elektronike dhe aktivitete të tjera, ku mbrojtja e përdoruesit ka një rëndësi të veçantë.

Për kriptografinë mund të thuhet se është një proces që përfshinë transformime të të dhënave ose mesazheve:

- shifrimi (enkriptimi);
- deshifrimi (dekriptimi).

**Shifrimi** ose **enkriptimi** është procesi i transformimit të një mesazhi në një shifër, dhe **deshifrimi** ose **dekriptimi** është procesi i kundërt që transformon një mesazh të shifruar (koduar) në formën e tij origjinale. Ky proces quhet proces kriptografik, dhe paraqitet në figurën si në vijim:



Figura 2: Procesi kriptografik

Shembulli më i njohur i kriptografisë është kodi i Morse-ut, i cili është përdorur në komunikimin telegrafik. Në figurën e mëposhtme është paraqitur secila shkronjë, numër dhe shenjë pikësimi të përfaqësuar nga kodi i Morse-ut:

	N O P Q R S H U V W X X		6 7 8 9 9	Plika Presja Dy plika Vizë Vijë e thyer Shenja për barazinë Plikëpyetja Shenja plus	(.) (.) (.) (.) (.) (.) (.) (.) (.) (.)
--	-------------------------	--	-----------------------	--	--

Por, shifrimi më shpesh i përdorur është kodi i Cezarit. Le të deshifrojmë, që do të thotë të drekriptojmë një porosi (mesazh) duke përdorur modelin e Cezarit!

# Detyrë

Jana dhe Azra janë shoqe nga klasa VII. Së bashku ata duhet të krijojnë një detyrë të projektit "Siguria në përdorimin e Internetit" dhe për atë qëllim ato kanë nevojë për Internet. Fjalëkalimi i hyrjes në internet është fshirë gjatë formatimit të kompjuterit. Vëllai i Janës i kujton fjalëkalimin, por dëshiron të luajë me ato, duke u dhënë atyre një fjalëkalim të shifruar dhe disa udhëzime: 1. deshifrimi mund të bëhet me modelin e Cezarit;

2. modeli i Cezarit përdor një numër sekret (çelës);

3. gjatë shifrimit, përdoren disa numra të plotë, siç çelësapër zhvendosje.

Fjalëkalimi	?	?	?	?	?	?	?	?	?	?	?
Çelësi	1	1	3	3	2	1	2	2	3	1	2
Shifra	H	М	С	L	Р	L	Y	R	F	J	Y

Le të fillojmë! (sqarim: çdo gjuhë ka alfabetin e vet, dhe shënohen të gjitha shkronjat në tabelë, nëse kërkohet për tu analizuar), në rastin tonë, do të shënojmë alfabetin (e gjuhës angleze) në një tabelë dhe pastaj do të përdorim çelësat. Në rastin konkret, jepen çelësat e deshifrimit, përkatësisht lëvizni për 1, 2 dhe 3 vende:

#### Tabela 1: Modeli i Cezarit me çelës 1

Shkronja	А	В	С	D	Ε	F	G	Н	I	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х	Y	Ζ
Çelësi	1	А	В	С	D	Ε	F	G	Н	Ι	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х	Y
Shifra	Ζ	А	В	С	D	Е	F	G	Н	I	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х	Y

#### Tabela 2: Modeli i Cezarit me çelës 2

Shkronja	А	В	С	D	Ε	F	G	Н	I	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х	Y	Ζ
Çelësi	1	2	А	В	С	D	Ε	F	G	Н	Ι	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х
Shifra	Y	Ζ	А	В	С	D	Ε	F	G	Н	Ι	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х

#### Tabela 3: Modeli i Cezarit me çelës 3

Shkronja	А	В	С	D	Ε	F	G	Н	I	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W	Х	Y	Ζ
Çelësi	1	2	3	А	В	С	D	Ε	F	G	Н	Ι	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W
Shifra	Х	Y	Ζ	А	В	С	D	Ε	F	G	Н	Ι	J	Κ	L	М	Ν	0	Ρ	Q	R	S	Т	U	۷	W

Në detyrë është dhënë shifra dhe çelësi, le të gjejmë fjalëkalimin!

Çelësi	1	1	3	3	2	1	2	2	3	1	2
Shifra	н	М	С	L	Р	L	Y	R	F	J	Y
Fjalëkalimi	I	N	F	0	R	М	Α	Т	I	K	Α



Mendimi logjik dhe zgjidhja e detyrave logjike konkurruese janë një hap përpara programimit. Programimi është procesi i përpunimit të një programi. Personat që krijojnë programe quhen programues. Detyrat logjike dhe zgjidhja e tyre kanë të bëjnë me konceptet informatike, siç janë: vargjet, listat, stekët/ grumbujt, rreshtat, rreshtat prioritar, numrat binarë, kriptografia, etj.



# TË PËRSËRISIM! TË USHTROJMË!



#### Detyra 1

Krijoni një varg të instruksioneve që tregojnë rrugën nga hyrja e shkollës deri në klasën tuaj.



#### Detyra 2

Hapni një fletë pune në MS Excel 2016. Emërtoni fletën e parë të punës "shtëpi" dhe në të, duke përdorur mënyra të ndryshme të përzgjedhjes, ngjyrosni qelizat ashtu që të fitoni shtëpi. Dokumentin e ruani me mbiemrin tuaj dhe klasën!



### Detyra 3

Numrat natyrorë 12, 18 dhe 36 shndërroni në numra binarë!



# Detyra 4

Numrin 236 e shndërroni në numër binar!



# Detyra 5

Krijoni porosi të shifruar duke përdorur modelin e Cezarit!



#### Detyra 6

Hulumtoni karakteristikat e lojrave Packman dhe Tetris. Me cilat koncepte informatike mund t'i lidhim? Shpjegoni!



### Detyra 7

Alma dëshiron të dërgojë mesazh te Ana: "DALIM" me kodin e Cezarit. Si duket mesazhi që Alma do ta dërgoj?



### Detyra 8

Stefani ka pranuar mesazh nga Amiri: "FMELKB" i koduar në kodin e Cezarit. Çfarë donte t'i thoshte Amiri Stefanit, nëse Stefani e di që çelësi është me numër tre (3)?



Jasmini i dërgoi një mesazh Tomit: "ACAU VSNY". Por mesazhi u përgjua nga Elvini. Elvini e di që Jasmini dhe Tomi korrespondojnë në një mënyrë të tillë, ku secila shkronjë e alfabetit zhvendoset nga gjashtë (6) vende. Kështu që ai e deshifroi mesazhin. Çfarë mesazhi mori Elvini gjatë deshifrimit?.

#### Detyra plotësuese:

Përtë praktikuartë menduarin logjik, në zgjidhjen e detyrave logjike konkurruese mund të përdoren manualët e detyrave dhe shpjegimin e detyrave nga Dabar, të cilat janë botuar në faqen zyrtare të internetit **www.talent.mk**.

Adresat e internetit për zgjidhjen e detyrave logjike konkurruese dhe lojërat që mund të përdoren janë në:

#### www.bebras.org

https://www.bbc.com/bitesize/articles/zqnc4wx

http://digit.mile.mk/



# Programimi i avancuar në mjedisin vizual

Hyrje në programim në mjedisin vizual Programimi grafik. Njohuri për programin Scratch Programet interaktive dhe ngjarjet Zhvillimi i programeve me situata problemore më komplekse LE TË PËRSËRISIM! LE TË USHTROJMË!

# 3. Hyrje në programim në mjedisin vizual

Të rikujtojmë!

Çfarë janë programet? Theksoni përkufizimet për programimin! Si përpunohet programi? Cilat gjuhë programuese i njeh?

Të menduarit logjik dhe zgjidhja e detyrave logjike konkurruese paraqet hyrje në programim. Nuk është rastësi që të menduarit logjik është një hap para **programimit**.

Prandaj, programimi mund të përkufizohet si një mënyrë ose aftësi e të menduarit në mënyrë që të arrihet në zgjidhjen përfundimtare të një detyre, të një problemi, ose situate problemore.



Figura 1: Logo-ja e Scratch

Si rrjedhim, qëllimi i programimit është krijimi i një bashkësie të udhëzimeve që kompjuteri duhet të ndjek për të kryer operacione specifike ose për të kryer veprimin e dëshiruar.

Kur shkruan dhe zhvillon programe, programuesi përdor një grup programesh, të cilat e përbëjnë **mjedisin e programimit**. Këto programe janë:

- editori ku shkruhet kodi burimor i programit;
- kompajleri e shndërron kodin burimor në instruksion të makinës;
- biblioteka e programeve të gatshme koleksion ose një grup programesh të vogla dhe shpesh të përdorura në mjedisin e programimit;
- debageri përdoret për gjetjen e gabimeve.

Programimi më interesant dhe tërheqës është programimi në mjedisin vizual, sepse gjatë zgjidhjes së situatave problemore, krijimit të hapave dhe instruksioneve, përdoren objekte vizuale, të tilla si: karaktere, prapavija, tinguj, lëvizje, efekte, etj, të cilat e bëjnë programimin të lehtë dhe argëtues.

Në procesin e të mësuarit të programimit ka disa aplikacione që janë shembuj të një ambienti vizual për krijimin e programeve, nga të cilët më i përdoruri është Scratch.

Scratch është një gjuhë programuese vizuale që krijon tregime interesante, animacione, lojëra, etj. Programimi në Scratch bazohet në bërjen e projekteve me ndihmën e objekteve, të ashtuquajturat figurat. Seti i tij i urdhërave të kujton në lojërat enigmatike (me bashkim formash), ato ofrojnë spektër të gjerë mundësish për krijimin e zgjidhjeve programuese më komplekse dhe mbizotërimin e problemit dhe të menduarit në mënyrën algoritmike.

Në këtë temë do të njihemi me **gjuhën e programimit Scratch**, me mjedisin e punës, dhe mënyrën e krijimit të aktiviteteve projektuese si zgjidhje të detyrave.



### Termat kryesorë!

paraqitje grafike, koordinata, konstante, ndryshore, deklarimi (urdhëra), programe interaktive, objekte, ngjarje (ndodhi).



# 3.1 Programimi grafik. Njohja me programin Scratch

Programimi në mjedisin vizual është një mënyrë interesante për të mësuar konceptin dhe hapat e krijimit të programit. Karakteristika kryesore të kësaj mënyre të programimit është që të gjitha mjetet, figurat, urdhërat (komandat) dhe madje kodet të **paraqiten grafikisht**. Ky interfejs i mundëson nxënësve të krijojnë projekte në mënyrën më të lehtë dhe të thjeshtë. Gjatë krijimit të projekteve, nxënësit zhvillojnë kreativitetin e tyre dhe mendojnë në mënyrë krijuese, fillojnë të mendojnë në mënyrë sistematike dhe të punojnë së bashku. Gjuha programuese që përdoret më shpesh dhe që implementohet në mjedisin visual për programim është **Scratch**.

Scratch është gjuhë programuese, me të cilin mund të krijoni tregime interaktive, animacione, lojëra, instrumente muzikore, etj. Meqenëse të gjitha mjetet paraqiten grafikisht nga ikonat e duhura, nxënësit mund të tërheqin dhe kombinojnë blloqe të kodeve, të përfshijnë efekte animacioni, efekte audio, etj, pa pasur nevojë të mësojnë sintaksën ose komandat e tekstit të gjuhës së programimit.

Mjedisi vizual i programimit **Scratch** mund të përdoret në internet (online) në linkun vijues: **https://scratch.mit.edu**. Kjo faqe është faqja zyrtare e **Scratch**-it. Së pari krijohet një llogari për përdoruesin, me të cilën përdoruesi paraqitet me emrin dhe fjalëkalimin e tij, pastaj fillon krijimi i projekteve. Ekziston edhe një version offline, dhe pa pagesë e që mund të shkarkohet nga faqja zyrtare.

#### 3.1.1 Startimi i Scratch

Versioni Offline i Scratch fillon me klikimin e dyfishtë në ikonën e tij, që është në sipërfaqen e punës **(Desktop)** të kompjuterit, pastaj hapet dritarja vijuese dhe qasemi në mjetet e punës:



Figura 1: Ekrani fillestar i Scratch

#### Le të përshkruajmë dritaren e punës në detaje:

- shiriti i titullit tregon emrin e programit;
- butonat e punës në dritare lejojnë të minimizoni, maksimizoni dhe mbyllni projektin;
- Mjeti i përzgjedhjes së gjuhës e lejon përdoruesin të zgjedhë se në cilën gjuhë të paraqiten menytë dhe urdhërat (komandat). Scratch është zhvilluar në shumë gjuhë;
- përmes menysë file përdoruesi kryen aktivitete me projektin: ruajtja, hapja e një projekti të ri, hapja e një projekti ekzistues dhe aktivitete të ngjashme;
- përmes shiritit të menysë kemi qasje në udhërat dhe rregullimin e objekteve që do të përdoren në projekt;
- në sipërfaqen e punës vendosen figurat (karakteret) të cilat janë figura në projekt;
- përmes pjesës për shtimin e figurave, përdoruesi mund të bëjë një përzgjedhje të karaktereve që do të jenë pjesë e detyrës së projektit;
- zgjedhja e prapavijës mundëson aplikimin e temës përkatëse në detyrën projektuese;
- paleta me urdhëra përman blloqe me urdhëra ose deklarime, të cilat përdoruesi mund t'i zgjedh. Sepse këto kanë funksion të ndryshëm, përkatësishtaplikohen për qëllime të ndryshme, për ngjyrosje të ndryshme. Përmes ngjyrosjes përdoruesi e di se cilës kategori i takon urdhëri;
- vendi i programit paraqet grupin e instruksioneve të cilat përdoruesi i ka zgjedhur dhe me tërhjekje i ka kombinuar për të paraqitur dhe kryer një aktivitet.

#### 3.1.2 Njohja me mjetet themelore të Scratch

#### **Objektet (Sprite)**

Kur përdoruesi fillon të krijoj një program, pasi të startoj me mjedisin vizual në ekran, së pari e vëren figurën ose karakterin që ka rol dominues në projekt, përkatësisht do të programohet për të kryer aktivitete. Në fillim karakteri është logoja e Scratch, por mund të ndryshohet duke klikuar në ikonën Choose Sprite From Library dhe zgjedhja nga galeria, siç tregohet në figurë:



Figura 2: Galeria e karaktereve në Scratch

Nga figura mund të shihet se karakteret ndahen në kategori të ndryshme. Objekti (**sprite**) shtohet duke klikuar në kategori, pastaj në objekt dhe në fund në butonin **OK**.

Objektet përveç nga galeria mund të shtohen nga kompjuteri, kamera ose nga karakter i krijuar nga përdoruesi.

#### Prapavija (Background)

Poashtu edhe prapavija vendoset në të njëjtën mënyrë si karakteret. Klikoni në ikonën e ndryshimit të prapavijës dhe shfaqet një dritare e përzgjedhjes së temës së prapavijës, siç tregohet në figurë:



Figura 3: Galeria me tema të prapavijës

Gjithashtu, temat e prapavijës mund të shtohen nga kompjuteri ose të krijohen vetë.

#### Urdhërat (deklarimet)

Në programim deklarimet janë urdhëra që i tregojnë kompjuterit se çfarë të bëjë. Në Scratch, deklarimet ndodhen në pjesën scripts:



Figura 4: Lloje të deklarimeve (urdhërave) në Scratch

Nga figura vërejmë se deklarimet, përkatësisht urdhërat ndahen në këto kategori: lëvizja, pamja, tingulli, lapsi, ngjarja, kontrolli etj. Secila kategori e ngjarjes ose urdhërit shënohet me ngjyrë të veçantë. Kodet e këtyre urdhërave janë në formën e blloqeve me ngjyra, si dhe kategoria.

Krijimi i një hapi programues bëhet duke klikuar dhe zvarritur bllokun dhe duke lidhur blloqet, përkatësisht hapat në një tërësi. Kur do të vendosen këto urdhëra, fillohet me programin e krijuar.

#### 3.1.3 Krijimi i programeve të thjeshta në Scratch

Për të kuptuar aftësitë dhe funksionalitetin e mjeteve Scratch, do të krijojmë një program të thjeshtë. Tema e detyrës së projektit është vallëzimi. Le të fillojmë!

Së pari e fillojmë programin Scratch dhe zgjedhim një karakter. Për këtë aktivitet të projektit zgjedhim karakterin Cassy Dance dhe e vendosim në mes të sipërfaqes së punës. Si ta gjejmë mesin e sipërfaqes së punës?



#### Mundohuni!

Me ndihmën e një mësuesi (arsimtari) të matematikës përpiquni të përcaktoni se çfarë është sistemi koordinativ? Si paraqitet?

Sipërfaqja e punës në **Scratch** përfaqësohet si një **sistem koordinativ** i përbërë nga boshtet x dhe y. Sistemi i koordinatave është në formën e rrjetit matematikor ose të skemës të përbërë nga shumë pika të quajtura **koordinata**. Vendndodhja e pikave në të vërtetë përcakton vlerat e x dhe y. Qendra ose mesi i sipërfaqes së punës ka vlerën: x = 0 dhe y = 0. Çdo pikë jashtë mesit ka vlerë të veçantë x dhe y. Në fakt, vlerat e koordinatave përcaktojnë vendin, përkatësisht pozicionin e objektit.

Karakteri **Cassy Dance** është një objekt në detyrën e projektit. Për ta vendosur në mes të sipërfaqes së punës, përmes menysë **motion** zgjedhim blloqet: **set x to 0** dhe **set y to 0**, përmes së cilës do të shënojmë vlerat e x dhe y me 0:



Figura 5: Ndarja e vlerave të x dhe y
Kur zgjedhim blloqet me një urdhër për vendosjen e vlerave të boshteve, vërejmë se vendi ku duhet të përcaktohet vlera është në formën e një rrethi, pra fushë për vendosje. Kjo na tregon se sipas nevojave, mund të ndryshojmë vlerën në mënyrë manuale sipas nevojave. Prandaj, në programim këto vende të rezervuara për vlera quhen **variabla** ose **ndryshore**.

Në fillim, objekti, në rastin tonë Cassy Dance, mund të jep mesazhe hyrëse. Këtë e bëjmë përmes menysë Scripts dhe opsionit Looks, ku zgjedhim blloqet komanduese të mëposhtme:

 say
 Përshendetje Unë jam Kejsi!
 for 2 secs

 switch
 costume
 to
 cassy dance-a

 say
 Eja të vallëzojmë
 for 2 secs

 switch
 costume
 to
 cassy dance-c

Figura 6: Seti i urdhërave për shkruarjen e porosisë

Nga kodi mund të vërehet se përveç se jepet urdhëri për porosi, ndryshimet bëhen ndryshime edhe në pozicionin e objektit, duke u shfaqur se karakteri lëviz. Për të filluar realizimin e setit të urdhërave, para setit të kodeve me ngjyrë vjollcë, përmes menysë **Events**, e shtojmë bllokun me urdhërin si në vijim:



Figura 7: Vendosja e fillimit për ekzekutimin e setit të instruksioneve

Me procedurën për vendosjen e **prapavijës (Backdrop)** zgjedhim **Spotlight Stage 2**. Ndërsa është zgjedhur prapavija, në shiritin e menysë zgjedhim **Backdrop** dhe fitohet kjo pamje:

Now backdrops		-(
New backdrop: backdrop1 460:00 apotlyht-stage2 450:05	backdropt  Clear 2445 2 3 dt	
	Bitmap Mode	

Figura 8: Rregullimi i prapavijës

Bëni një kopje të **backdrop 1** duke klikuar me të djathtën dhe opsionin **duplicate**. Paraqitet edhe një prapavijë e bardhë. Këto dy prapavija mund t'i rregullojmë duke zgjedhur mjetin **Fill with color**, zgjedhim ngjyrën nga paleta me ngjyra e cila gjendet në pjesën e poshtme të dritares dhe e ngjyrosim prapavijën. Të njëjtën e përsërisim edhe për **backdrop 2**.



Figura 9: Krijimi dhe rregullimi i prapavijës

Këto prapavija të shtuara, të krijuara dhe rregulluara vendosen në kodin e aktivitetit të projektit. Përmes menysë **Look**, i shtojmë blloqet me kode për të ndryshuar prapavijën. Në vijim është dhënë radhitja e urdhërave:



Figura 10: Seti i urdhërave për ndryshim të prapavijës në aktivitetin e projektit

Për të vendosur objektin në skenë, u japim vlera koordinatave, përmes menysë **motion**, komandat set x to 15 dhe set y to 100. Objekti i vendosur në këtë mënyrë do të programohet për të kryer vallëzim dhe pastaj do të shoqërohet me muzikë. Bllok-urdhërat, përkatësisht deklarimet që duhet të shtohen janë në rendin si në vijim:

P
x: 15 y: 100
when 庵 clicked
set x to 0
Set y to to
when is clicked
switch backdrop to backdrop1
say Përshëndetje! Unë jam Kejsi! for 2 secs
switch backdrop to backdrop2
switch costume to cassy dance-b
say Eja të vallëzojmë! for 2 secs
switch costume to cassy dance-c
switch costume to cassy dance-c switch backdrop to spotlight-stage2
switch costume to cassy dance-c switch backdrop to spotlight-stage2 wait 1 secs
switch costume to cassy dance-c switch backdrop to spotlight-stage2 wait 1 secs set x to 15
switch costume to cassy dance-c switch backdrop to spotlight-stage2 wait 1 secs set x to 15 set y to 100
switch costume to cassy dance-c * switch backdrop to spotlight-stage2 * wait 1 secs set x to 15 set y to 100 repeat 10
switch costume to cassy dance-c switch backdrop to spotlight-stage2 wait 1 secs set x to 15 set y to 100 repeat 10 next costume
switch costume to cassy dance-c * switch backdrop to spotlight-stage2 * wait 1 secs set x to 15 set y to 100 repeat 10 next costume play sound hip hop * until done

Figura 11: Shtimi i cikleve për përsëritje

Për të kompletuar kodin shtohet seria të instruksioneve. Për shembull, detyrën tonë projektuese do ta përfundojmë me porosi dhe ndalimin e të gjitha proceseve:

wait 1 secs	
say Faleminderit!	for 2 secs
stop all	

Figura 12: Përfundimi i programit

Me këtë kemi përfunduar projektin tonë të parë. Le të hedhim një vështrim!

Në anën e majtë të dritares së punës, në pjesën ku shfaqen objektet, në këndin e sipërm të djathtë, ka ikona për fillimin dhe përfundimin e projektit:



Figura 13: Butonat për fillimin ose përfundimin e programit

Duke klikuar në flamurin e gjelbër fillojmë të ekzekutojmë grupin e instruksioneve, ku shohim vizualisht se çfarë kemi bërë, nëse duam të përfundojmë përsëri paraqitjen, klikojmë në butonin e kuq.



Programet për programim grafik mundësojnë krijimin e vargjeve të urdhërave për të ekzekutuar një program përmes përdorimit të mjeteve, figurave, komandave, etj., të cilat kanë një paraqitje grafike. Scratch është një gjuhë programimi i cili mundëson të krijoni tregime interaktive, animacione, lojëra, etj. Komandat e Scratch shfaqen në blloqe që duken si enigma. Ato kombinohen me tërheqje dhe bashkim. Sipërfaqja e punës në Scratch mund të përfaqësohet si rrjet matematikor ose skemë e përbërë nga pikat-koordinatat. Objekteve të shtuara në detyrën e projektit në Scratch, u caktohen blloqe deklarimesh (urdhëra). Deklarimet (urdhërat) i tregojnë kompjuterit se çfarë të bëjë. Disa deklarime përmbajnë vende të rezervuara për vlera dhe ato quhen variabla ose ndryshore.



- 1. Çfarë paraqesin blloqet në Scratch? Si duken ata?
- 2. Si ndahen bllok-urdhërat? Theksoni disa kategori!
- 3. Çfarë do të thotë flamuri i gjelbër që gjendet në pjesën lartë të paraqitjes së projektit?
- 4. Çfarë na tregojnë koordinatat e objektit në detyrën e projektit?
- 5. Si vendoset prapavija në detyrën projektuese?



#### 3.2 Programet interaktive me ngjarje

### Të rikujtojmë!

A ju kujtohet kur përmendëm për herë të parë termin "programe interaktive (ndërvepruese)"? Në cilat tema të përpunimit? A mund të mbani mend se si e përkufizuam atë?

Termi "**interaktiv**" theksohet shumë herë në tema të ndryshme dhe në fusha të ndryshme. Për shembull, orën e informatikës mund ta karakterizojmë si **interaktive**, sepse ka aktivitete, komunikim për realizimin e aktiviteteve. Shpeshherë bëhen diskutime për zgjedhjen e detyrave dhe situatave problemore, që do të thotë që të gjithë mendojmë, japim përfundime, pikëpamje, por edhe përgjigje për shumë pyetje.

E njëjta gjë ndodh kur përdorim softuerin arsimor. Programi jep instruksione, dhe ne si përdorues i respektojmë instruksionet dhe në përputhje me këtë, ndërmarrim aktivitetet e duhura.

Kur bëhet fjalë për lojërat kompjuterike, ndjekja e rregullave të lojës në të vërtetë përfaqëson këtë interaktivitet (ndërveprim) ndërmjet përdoruesit dhe vetë lojës. Çdo përdorues dëshiron të arrijë qëllimin, pra të fitojë dhe për këtë arsye duhet të veproj në përputhje me rregullat e lojës.

Në këtë njësi mësimore do të krijojmë një projekt që përfshinë ngjarje me **karakter interaktiv**.

Karakteri kryesor në lojën tonë interaktive është **Dinoaur 1**. Ai do të kryejë llogaritjet matematikore të detyrave që do të caktojmë. Për fillim, do të krijojmë vargje të bllok-urdhërave që do të llogaritin herësin e dy numrave të dhënë.

Le të fillojmë me programimin!

Së pari ne fillojmë programin **Scratch**. Me ndihmën e mjetit gërshërë (**delete**) e fshijmë objektin, përkatësisht karakterin **Cat**. Përmes mjetit për shtimin e objekteve, në pjesën **spirit**, shtojmë një objekt nga galeria ekzistuese. Për projektin tonë do të zgjedhim objektin **Dinoaur 1**. Pastaj, shtojmë prapavijën (**backdrop**). Prapavijën do ta krijojmë dhe rregullojmë me ndihmën e mjeteve dhe paletës së ngjyrave të ofruara nga programi, dhe prapavija e dytë është një fotografi nga kompjuteri ynë. Ekrani bazë do të dukej kështu:



Figura 1: Shtimi i objektit dhe prapavijës së projektit

Klikojmë në Dinoaur 1 dhe fillojmë me krijimin e serisë së bllok-urdhërave (komandave):



Figura 2: Përcaktimi i vendit të objektit

Nga sekuenca e specifikuar e bllok-urdhërave, me klikim në flamurin e gjelbër që tregon fillimin e ekzekutimit të sekuencës së instruksioneve, vlera e x dhe y do të jetë -122 dhe -85, që do të thotë se kemi përcaktuar lokacionin e Dinoaurl.

Bllok kodet e mëposhtme janë ngjarjet fillestare në projekt:

when 🚈 clicked										
say Përshëndetje matematikan for (2) sec										
glide 1 secs to x: 0 y: 0		1	1							
think A mund të ndihmojmë shokun që të zgjid	dh disa detyra	matemati	kore?					for	2	secs
change color effect by 25				. 1	-					
change color effect by 25 switch costume to dinosaur1-c							-			
change color effect by 25 switch costume to dinosaur1-c say Letëprovojmë for 2 secs										

Figura 3: sekuenca nga bllok-urdhërat që shfaqin porosi

Kjo do të thotë që kur të klikojmë në flamurin e gjelbër, do të ekzekutohet seria e bllok-urdhërave. Para së gjithash, **Dinoaur 1** na përshëndet me "**përshëndetje matematikan**", lëviz, përkatësisht "**rrëshqet**" deri në mesin e sipërfaqes së punës. Kur **Dinoaur 1** vin në mes të të sipërfaqes së punës, duket sikur mendon, ndryshon ngjyra e tij dhe paraqitja që ngjanë në lëvizje. Pastaj, vazhdojmë me serinë e blloqeve që do të na çojnë në një situatë të re. **Dinoaur 1** do të jep instruksionet dhe ne nga tastiera do të vendosim të dhënat e kërkuara. Në fakt, **Dinoaur 1** është matematikani ynë dhe ai do të na tregojë rezultatin e llogaritjes së herësit të dy numrave të dhënë. Për këtë qëllim, së pari do të përcaktojmë tre variabla, përkatësisht variablat që do të kenë vlera të futura nga tastiera dhe rezultatin e llogaritjes. Procedura për krijimin e variablave është si në vijim: përmes menysë **Scripts** hyjmë në kategorinë **Data** dhe aty klikojmë në **Make variable**. Meqenëse këtë variabël do ta përdorim në situata të tjera në aktivitetin e projektit, e klikojmë opsionin dhe e emërtojmë variablin.

# Shënim!

emri i variablit, përkatësisht ndryshores, mund të jepet sipas dëshirës, por duhet të mendohet në atë çfarë vlere do duhet të ketë dhe cili është roli. Për shembull, në detyrë kemi përcaktuar variablat me emrat Numër 1 dhe Numër 2, por ato gjithashtu mund të përcaktohen si i pjesëtueshmi dhe pjesëtuesi sipas kuptimeve matematikore.

Scripts	Costumes	Sounds
Motion	Ever	nts
Looks	Con	trol
Sound	Sen	sing
Pen	Ope	rators
Data	More	Blocks
Numri 2 Herë	si	
set Here		
change	Herësi 🔹 by	1
show var	<b>iable</b> Herësi	•
hide vari	able Herësi	•

Figura 4: Krijimi i variablave/ndryshoreve

Në detyrën konkrete të projektit krijojmë **tre variabla**. Ndryshorja e parë do të përmbajë vlerën që e japim nga tastiera për numrin e parë (**numri 1**), ndryshorja e dytë do të përmbajë vlerën për numrin e dytë (**numri 2**) dhe ndryshorja e tretë (**herësi**) do të shfaq vlerën që rezulton nga zgjidhja. Le t'i zbatojmë në mënyrën përkatëse variablat e krijuara!

set x to -122			
set y to -85			
wait 1 secs			
switch backdrop to blue sky2 -			
say Të japim dy numra dhe të gjejmë herësin e tyre	for	2	secs
switch costume to dinosaur1-c			1
ask Vendose numrin e parë and wait			
set Numri 1 • to answer			
show variable Numri 1			
switch costume to dinosaur1-f			
ask Vendose numrin e dytë and wait			
set Numri 2 v to answer			
show variable Numri 2 💌			

Figura 5: Aplikimi i bllok-urdhërave nga kategoria Data

Përmes menysë **Motion** u japim vlera **x** dhe **y**, në mënyrë që ta kthejmë **Dionaur 1** në pozicionin e tij origjinal, pra në pozicionin fillestar. Përmes menysë **Looks**, shtohet bllok-urdhër për të ndryshuar prapavijën (**backdrop**) dhe një bllok-urdhër për të paraqitur një porosi. **Dinoaur 1** ndryshon pamjen tij dhe kërkon të vendosim numrin e parë. Këtu zbatohet bllok-urdhëri, përmes të cilit mundësohet opsioni për vendosjen e numrit nga tastiera dhe shfaqjen në fushën me vlerën e vendosur. Përsëriteni të njëjtën gjë për numrin e dytë, siç paraqitet në figurë.

Diçka tjetër që duhet të bëhet është të kontrollohet nëse vlera e numrit të dytë është 0, sepse nga rregullat e matematikës dihet se pjesëtimi me zero nuk është i mundur. Për këtë qëllim, përmes menysë **Control** aplikohet bllokurdhëri: "nëse kushti është i plotësuar, atëherë ekzekutoni instruksionet si në vijim" dhe "nëse kushti nuk është i plotësuar, ekzekutoni instruksionet si në vijim:

	1								
	<u></u>	Numri 2		then					
	say	Pjesëtimi me z	ero nuk ësh	të i mundi	ur! f	or (	2	sec	5
-	swite	ch costun	ne to	dinosa	ur1-	g 🔻			
els	•								2
-	set	Herësi	• to	Numri 1		1	Num	ri 2	)
	shov	v variable	Heri	ėsi 🔻		-			
1	saγ	Herësi i numra	ve është:	for	2 5	ecs			
-	say	Herësi	<b>for</b>	<b>2</b> s	205				
	swite	ch backdı	op to	3 -					
	hide	variable	Herësi	-					
	hide	variable	Numri 1	-					
	hide	variable	Numri 2	-					
	~								
_	-								

Figura 6: Aplikimi i bllok-urdhërit If/Else

Ose, nëse vlera e numrit të dytë është e barabartë me zero, paraqitet porosia: "pjesëtimi me zero nuk është i mundur" dhe **Dinoaur 1** ndryshon paraqitjen. Nëse kushti nuk është i plotësuar atëherë:

- të llogaritet herësi;
- të paraqitet vlera e herësit;
- herësi të paraqitet si porosi e Dinoaur I;
- të kalohet në temë të re, përkatësisht prapavijë;
- kur të kalohet në prapavijën e re, variablat nuk duhet të jenë të dukshme.

Me këtë kemi përfunduar llogaritjen e herësit të dy numrave. Projekti mund të përfundoj sipas dëshirës. Për shembull, në situatën tonë **Dinoaur 1** vazhdon me mesazhe:

wait 1 secs	
switch costume to dinosaur1-a	
say Vazhdojmë mëtutje for 2 secs	
think Hmmm for 2 secs	
say Vazhdoni me zgjidhjen e detyrave, mbledhje, zbritje, shumëzimi for 2 secs	
switch costume to dinosaur1-d	
say E di që do ti ndiqni instruksionet for 2 secs	
switch costume to dinosaur1-c	
say Provoni! Ju dëshiroj suksese! for 1 secs	
switch backdrop to blue sky2	
stop all T	

Figura 7: Bllok-urdhërat për paraqitje të porosive

Procedurën për paraqitjen e porosive në ekran e zbatojmë paraprakisht. Porositë varen nga kreativiteti i përdoruesit dhe nga mënyra se si ai e mendon vizualisht zgjidhjen e detyrës së projektit. Me urdhërin "**Stop all**", shënohet fundi i ekzekutimit të sekuencës së urdhërave.

Duke klikuar në flamurin e gjelbër fillojmë të ekzekutojmë sekuencat e bllok-urdhërave, dhe kështu kontrollojmë atë që kemi bërë. Ky projekt është interaktiv, sepse Dinoaur 1 jep instruksione, tregon se si të veprojmë dhe si të ekzekutojmë këto instruksione duke realizuar veprime nga tastiera.



# Detyrë

Dinoaur 1 na ndihmoi të gjejmë herësin e dy numrave të dhënë. Vazhdoni detyrën, ashtu që të krijoni vargje të bllok-urdhërave që do të llogaritin operacione të tjera matematikore, si mbledhja e dy numrave, zbritja e dy numrave dhe prodhimi.



A është e mundur të krijojmë një sekuencë të bllok-urdhërave (bllokkomandave) që do të kryejnë llogaritjet e operacioneve matematikore me tre numra? Çfarë mungon në programin Dinoaurl që u krijua për të kryer llogaritjet duke futur dy numra? Shpjegoni procedurën! Tregoni se ku do të bënit ndryshime!



Programet ku përdoruesi mund të vendos vlerat e variablave me të cilat paraqiten rezultatet dhe varen nga njësitë hyrëse quhen programe interaktive (ndërvepruese). Variablat janë ndryshore që marrin vlera kur vendosen të dhëna nga tastiera ose, si rezultat i llogaritjes. Variablat krijohen përmes menysë Scripts dhe kategorisë Data.



- 1. Cilat programe quhen interaktive?
- 2. Çfarë janë variablat? Cila është procedura për krijimin e variablave?
- 3. Cili urdhër lejon të vendosjen e vlerës së ndryshores?
- 4. Si e vendos përdoruesi vlerën e ndryshores në program?
- 5. Pse shërben urdhëri (komanda) If ... Else?



# 3.3 Përpunimi i programeve me situata problemore më komplekse

Krijimi i lojës në Scratch është punë pak më komplekse, sepse përfshinë shumë objekte dhe ngjarje. Secilit objekt i jepet instruksione ose set (grup) instruksionesh për të arritur qëllimin, përkatësisht deri në fitoren përfundimtare në lojë. Pastaj, ngjarjet që mund të ndodhin duhet të parashikohen. Sigurisht, pritet që atyre tu ofrohen zgjidhje të përshtatshme. Për shembull, nëse fitohet, të merret mesazh ose pra, të dëgjohet tingull fitues dhe nëse humb ta kthej objektin në fillim.

Për shembull, do të krijojmë një lojë që ka dy objekte: **Cat** dhe **Gobo**. **Cat** do të lëvizë sipas lëvizjes së mausit - majtas dhe djathtas. Më shumë **Gobo** bien nga qielli, dhe **Cat** duhet të mbledhë më shumë nga numri i përgjithshëm i dhuratave.

Le të fillojmë një projekt të ri në Scratch.

Duke klikuar në butonin **Choose Sprite from library**, shtojmë një objekt/ karakter të ri. Nga galeria me objekte/karaktere zgjedhim Cat dhe Gobo:



Figura 1: Objekte/karaktere nga projekti

Duke përdorur mjetin **Shrink**, i ndryshojmë dimensionet e objekteve/ karaktereve. Klikojmë në mjetin Shrink, dhe pastaj klikojmë në objekt, përkatësisht në karakterin, sa herë që duam të zvogëlojmë objektin. Në rastin tonë do të zvogëlojmë dimensionet e të dy objekteve:



Figura 2: Ndryshojmë dimensionet përmes mjetit Shrink

Shtojmë temë ose prapavijë përmes mjetit **choose backdrop from library**. Për lojën tonë do të përdorim temën **Gingerbread**:



Figura 3: Tema e projektit

Për secilin objekt duhet të caktohen set-e ose sekuenca prej instruksioneve për të mundësuar që objektet të animohen dhe të kryejnë disa veprime. Për shembull, objekti **Sprite 1** duhet të lëvizë majtas - djathtas dhe të mbledhë **Gobo**:



Figura 4: Set-i i instruksioneve për lëvizjen e Sprite 1

Me urdhëra t'i shpjegojmë bllok për bllok!

Klikimi në flamurin e gjelbër do të fillojë procesin e ekzekutimit të instruksioneve të mëposhtme. Në këtë mënyrë, u jepen vlera koordinatave **x** dhe **y**, **0** dhe -130. Shënojmë drejtimin e lëvizjes 90 djathtas. Objekti/karakteri do të lëvizë përgjatë **boshtit x** sipas lëvizjes së mausit. Gjithashtu, duke klikuar në flamurin e gjelbër do të fillohet me emitimin e fajllit me zë (audio file). Në rastin tonë nga galeria me tinguj është zgjedhur zëri dance **slow mo**.

Gjithashtu, instruksione duhet t'i japim edhe objektit/karakterit **Gobo**. Sipas detyrës tonë, nga qielli teposht duhet të bien më shumë ekzemplarë ose **klone të Gobo**. Megjithëse kemi përcaktuar vetëm një objekt - dhuratë, përsëri gjatë lojës ky objekt do të numërohet. Le të shohim procedurën:



Figura 5: Krijimi i kloneve në objekt

Kjo sekuencë ose set i urdhërave do të thotë që me klikim në flamurin e gjelbër shënohet fillimi i ekzekutimit të sekuencës së komandave për të krijuar një klon, pra një kopje të objektit/karakterit. Me bllok-urdhërin repeat lejohet të krijoni klone në çdo sekondë, derisa të arrihet vlera 100.

Diçka tjetër që duhet të bëhet është të krijohen blloqe me urdhëra që do t'i tregojnë objektit - klonit si të veproj dhe çfarë veprimesh të kryej:

g S	prit	e1	٠	?	2
			-	-	-
	g S	g Sprit	g Sprite1	g Sprite1	g Spritel ?

Figura 6: Instruksionet për objektin-klon

Kjo do të nënkuptojë se: kur do të paraqitet objekti i klonit, ai do të lëvizë poshtë deri te objekti **Spirit 1**. Kur të prek **Spirit 1** do të fillojë të emetoj një fajll muzikor (music file) të quajtur **zoop**. Me urdhërin ciklik të përsëritjes – **repeat** do të ndryshojnë katër format e ndryshme të objektit Gobo në çdo sekondë. Menjëherë pas kësaj, do të fshihet, përkatësisht nuk do të jetë më objekti-klon.

Set-i (grupi) tjetër i instruksioneve tregon se objekti i klonit do të shfaqet në koordinatat arbitrare në intervalin prej -200 deri 200 në lidhje me **boshtin x** dhe do të lëvizë poshtë duke filluar nga vlera 140 në raport me **boshtin y**. Gjatë kësaj, do të ndryshoj paraqitjen e tij derisa të prek objektin **Spirit 1**. Në fund objekti-klon do të humb.



Figura 7: Vendosja e koordinatave arbitrare për paraqitjen e objekti-klon

Në lojë na mungojnë pikët. Me serinë e ardhshme të komandave do të mundësojmë numërimin. Kur **Gobo** prek objektin **Spirit 1**, atëherë do të ndryshojnë pikët. Së pari, duhet të shtojmë ndryshore të re që fillimisht do të ketë një vlerë 0, dhe pastaj do të rritet për 1. Shtimi i ndryshores bëhet duke klikuar në opsionin **Data** nga menyja **Scripts**, pastaj në opsionin **Make variable**:



Figura 8: Krijimi i ndryshore së re

Paraqitet dritarja vijuese, përmes së cilës i jepet emri variablës (ndryshores), dhe tregon se variabla e krijuar do të përdoret për karakterin e selektuar:

	New	Variable
Variable na	me:	
For all s	prites	O For this sprite only
	OK	Cancel

Figura 9: Përkufizimi i variablës (ndryshores) së re

Me klikim në OK, shtohen bllok-urdhërat vijues të cilët i referohen ndryshores së re:

Scripts	Costumes	Sounds
Motion	Eve	nts
Looks	Cor	ntrol
Sound	Ser	ising
Pen	Ope	erators
Data	Mor	e Blocks
set Score	e • to 0	1
change show var	iable Score	
hide vari	able Score *	
Make a Li	ist	

Figura 10: Bllok-urdhërat të cilët i referohen ndryshores së re Score

Pas krijimit të variablës, krijojmë një grup të bllok-urdhërave të cilët duhet të ekzekutohen, për të numëruar pikët:

whe	n / clicked
set	Score to 0
fore	touching Gobo ? then
	change Score • by 1
	play sound chomp " until done

Figura 10: Bllok-urdhërat të cilët i referohen ndryshores së re Score

Le të shohim se si funksionon loja! Klikojmë në flamurin e gjelbër dhe fillojmë me lojën.



Krijoni një lojë të thjeshtë në Scratch. Lojërat që i keni krijuar i ndani me miqtë tuaj dhe luani duke ndjekur instruksionet.



Projektet që përfshijnë shumë objekte dhe ngjarje janë më komplekse për tu krijuar. Secili objekt është i programuar në veçanti, pra i jepen sekuenca të bllok-urdhërave për të ekzekutuar me qëllim krijimin e lëvizjes, aktiviteteve ose veprimeve. Në programe, objektet përshkruhen nga gjendja dhe mënyra e sjelljes. Në botën reale, objekt mund të jetë: makina, biçikleta, topi ose një karakter i përcaktuar nga një gjendje dhe sjellje. Ngjarjet ndodhin në varësi të gjendjes ose sjelljes së objektit. Këto gjëra duhet të parashikohen dhe duhet të ofrohet një zgjidhje e përshtatshme.



- 1. Çfarë janë objektet?
- 2. Si programohen objektet?
- 3. Kush është lidhja midis objekteve dhe ngjarjeve?



# TË PËRSËRISIM! TË USHTROJMË!



## Detyra 1

Mundohuni të krijoni një detyrë të projektit në Scratch, duke zbatuar blloqet e deklarimeve (komandave) që praktikisht i kemi punuar në klasë. Vendosni prapavijë, shtoni objekt, lëvizni objektin, shtoni zërin, etj..



# Detyra 2

Krijoni një lojë të thjeshtë në Scratch që përfshinë më shumë objekte dhe ngjarje.



# Detyra 3

Krijoni programe të veçanta që do të llogaritin sipërfaqen dhe perimetrin e një katrori, drejtkëndëshi dhe rrethi.



## Detyra 4

Krijoni një projekt në Scratch të natyrës interaktive, ashtu që kur të jepni një numër nga tastiera, të kontrolloj nëse numri është pozitiv apo negativ.

# Programimi përmes gjuhës programuese standarde të strukturuar

Hyrje në programim përmes gjuhës programuese standarde të strukturuar Procesi i përpunimit (zhvillimit) të një programi Njohja me elementet themelore të mjedisit të integruar për programim Pamja e shembullit të gatshëm të kodeve programore Ekzekutimi i programeve me shembuj të gatshëm Bazat e elementeve të gjuhës programore C++ Deklarimet Përpunimi i programeve Operacionet matematikore dhe shprehjet Konstantet dhe ndryshoret Deklarimet (teknikat) për vendosjen e të dhënave në program Shprehjet krahasuese

Struktura për përsëritjen në cikël deri në plotësimin e kushtit TË PËRSËRISIM! TË USHTROJMË!

# 4. Hyrje në programim përmes gjuhës programuese standarde të strukturuar

#### Detyra 1

Të punojmë në mjedisin për programim Code: Blocks dhe të krijojmë program në gjuhën programuese C++.

Jemi duke jetuar në një kohë të ndryshimeve të shpejta dhe të zhvillimit edhe më të shpejtë teknologjik në të gjitha pjesët e shoqërisë sonë. Pothuajse është e pamundur të mendohet ndonjë punë që do të bëhej pa përdorimin e kompjuterit. Duke pasur parasysh faktin se një kompjuter është një pajisje elektronike që përdoret për të mbledhur, ruajtur dhe përpunuar të dhëna me ndihmën e programeve të instaluara në të, konkludojmë se kompjuteri është pajisja më e fuqishme që ofron një zgjidhje për çdo situatë problemore, për qëllime të ndryshme dhe nga fusha të ndryshme.

Në përditshmërinë tonë përdorim programe të ndryshme, si për shembull:

- luajmë lojëra për zbavitje;
- shkruajmë ese në programin për përpunim të tekstit (MS Word, Open Office Writer etj.);
- kryejmë llogaritje në programin për llogaritje tabelare (MS Excel, Open Office Calc etj.);
- krijojmë prezantime në programin për krijimin dhe rregullimin e prezantimeve multimediale (MS Power Point, Open Office Impress, Prezi etj.);
- shikojmë videofajlla përmes programeve përkatëse (Winamp, VLC Player, Real Player etj.);
- krijojmë dhe rregullojmë figura (imazhe) me editorë për grafikë (MS Paint, InDesign, Corel Draw, Photoshop etj.);
- komunikojmë me të afërmit dhe miqët përmes programeve për komunikim (Viber, Messenger, WhatsApp etj.).

Ekzistojnë programe të tjera për qëllime specifike që përdoruesi mund t'i porosis, ose të krijohen në përputhje me kërkesat. Në fakt, programet janë lidhja ndërmjet përdoruesit dhe kompjuterit.

Prandaj, programet kompjuterike janë një seri instruksionesh që duhet t'i ekzekutoj një kompjuter. Procesi i dizajnimit, krijimit dhe lidhjes së vargjeve të instruksioneve quhet programim. Njerëzit të cilët krijojnë programe duke përdorur gjuhë programuese quhen programues. Gjuha programuese paraqet bashkësi të rregullave, simboleve dhe fjalëve kyçe, të cilët kombinohen në mënyrë të ndryshme me qëllim të arritjes deri te zgjidhja e detyrës ose situatës problemore. Të mësuarit e programimit e zhvillon mënyrën logjike të të menduarit, që është e shfrytëzueshme dhe e zbatueshme në situatat e përditëshme.



#### Termat kryesorë!

përkthyes, program burimor, program ekzekutues, editor, kompajler, debager, deklarim, shprehje krahasuese, cikël.



## 4.1 Procesi i përpunimit të një programi

Krijimi i një programi, qoftë ai për kompjuter, telefon, tablet ose pajisje tjetër, është një proces kompleks, sepse për krijimin e programit, është shfrytëzuar gjuhë programuese. Ka shumë gjuhë programuese dhe secila ka një grup të veçantë të instruksioneve dhe sintaksën për të shkruar programin. **Gjuhët e programuese** zakonisht ndahen në gjuhë programuese **të larta** dhe **të ulëta**.

**Gjuhët e ulëta të programimit** janë të orientuara drejt makinës, sepse ato varen nga makina në të cilën ekzekutohen, që do të thotë se nëse një program është shkruar në një procesor, mund të mos funksionoj në një procesor tjetër. Prandaj, programimi me kodin e makinës është i vështirë, sepse duhet të njihni mirë strukturën dhe ndërtimin e kompjuterit.

Për dallim nga gjuhët e ulëta programuese, **gjuhët e larta programuese** përdoren gjithnjë e më shumë, sepse ato nuk varen nga makina në të cilën ato kryhen, që do të thotë se ato mund të kryhen në kompjuterë me procesorë të ndryshëm. Nga ana tjetër, simbolet, shenjat e tyre, por edhe sintaksa e tyre është shumë e ngjashme me gjuhën natyrore dhe janë të lehta për t'i kuptuar. Gjuhët e larta programuese më të përdorura janë: **C**, **C++**, **Fortran**, **Basic**, **Pascal**, **Java** dhe etj. Kur krijojmë një program, pavarësisht se çfarë lloj i gjuhës programuese përdorim, procesi vazhdon të kalojë nëpër **fazat** e mëposhtme:

Nr. rendor	Faza	Kuptimi
1	Shkruarja e kodit burimor	Shkruarja e kodit burimor kryhet me gjuhë programuese në mjedisin e integruar për programim. Pastaj, ky program quhet <b>kod burimor (Source</b> <b>code)</b> që ruhet në pajisjet memorike të kompjuterit, si fajll me prapashtesën <b>.cpp</b> , për shembull: Ushtrimi.cpp
2	Përkthimi i kodit burimor	Kodi burimor me ndihmën e programit për përkthim që quhen <b>kompajler</b> , i përkthejnë në kod burimor. Gjatë përkthimit të kodit burimor mund të paaraqiten gabime. Këto gabime quhen gabime sintaksore dhe më shumë i përket gabimeve të fjalëve të gabuara nga gjuha programuese, shenja të shkruara gabimisht ose shenja të haruara të interpunkcionit.
3	Lidhja e kodi ekzekutiv	I lidhin objekt kodet me kodin <b>ekzekutiv (Executable code)</b> . <b>Kodin ekzekutiv</b> e realizon kompjuteri. Pastaj, krijohet fajlli ekzekutiv që ka prapashtesën <b>.exe</b> , për shembul: Ushtrimi.exe
4	Testimi i programit	Testimi i programit shërben për verfikimin të programit për atë se a i plotëson kushtet e vendosura dhe vinë deri te zgjidhja e saktë, përkatësisht paraqet një lloj hulumtimi që siguron informacione për kualitetin e prodhimit dhe eventualisht gjetjen e gabimeve. Kjo fazë paraqet edhe procesin e validimit dhe verifikimit të programit, që do të thotë i plotëson kërkesat e theksuara në dokumentacion, punon sipas pritshmërisë dhe mund të jetë e implementuar me të gjitha karakteristikat e tyre.

Tabela 1: Faza gjatë shkruarjes së programit

Mundohuni!

Kujtohu në programin Offline Scratch. Bëni krahasimin se ku dhe si i keni shkruajtur kodet, si kompajlohen dhe si gjenden gabimet gjatë krijimit të programeve. Theksoj ngjashmëritë dhe dallimet.

Në procesin e krijimit të programit në këtë njësi tematike do të krijojmë programe me gjuhën programuese C++ në Code::Bllock në mjedisin e integruar të programimit.



Programet kompjuterike janë varg instruksionesh që i ekzekuton kompjuteri. Procesi i krijimit të një programi quhet programim. Njerëzit të cilët krijojnë programe duke përdorur gjuhë programuese quhen programues. Gjuhët e programimit ndahen në gjuhë të ulëta dhe të larta. Fazat gjatë krijimit të një programi janë: shkrimi i kodit burimor, përkthimi i kodit burimor, lidhja me kodin e ekzekutueshëm, testimi dhe verifikimi i programit.



- 1. Si quhet procedura për krijimin e programeve?
- 2. Kush i krijon programet? Si?
- 3. Si ndahen gjuhët programuese?
- 4. Cilat janë gjuhët e ulëta programuese?
- 5. Numëro disa gjuhë të larta programuese?
- 6. Kush është ndryshimi midis gjuhëve të larta dhe të ulëta programuese?
- 7. Përkufizoni se çfarë është sintaksa e gjuhës programuese!
- 8. Çfarë është një program burimor? Çfarë prapashtese ka fajlli i programit burimor?
- 9. Si quhet programi që ekzekuton kompjuteri?
- 10. Numëroni fazat në krijimin e një programi!

# 4.2 Njohja me elementet themelore të mjedisit të integruar për programim

**Mjedisi i integruar për programim (IDE – Integrated Development Environment)** është një paketë softuerike, që përmban mjetet themelore të nevojshme për të shkruar dhe testuar softuerin. Ky aplikacion zakonisht ofron shumë funksione për shkruarjen, modifikimin, editimin, vendosjen dhe debagimin (korrigjimin) e programeve. Qëllimi kryesor i kësaj pakete është të thjeshtoj zhvillimin e softuerit dhe të minimizoj gabimet e kodimit.

Gjatë krijimit të programit, përdoruesit shfrytëzojnë një numër të caktuara mjetesh për ndërtimin dhe testimin e kodit softuerik. Mjetet e zhvillimit shpesh përfshijnë **redaktuesit (editorët) të kodit burimor, bibliotekat e kodeve, kompajlerët dhe platformat e testimit**. Në fakt, programuesi shkruan dhe editon kodin burimor (Editor), kompajleri përkthen kodin burimor në gjuhë të lexueshme, ashtu që kompjuteri të mund ta ekzekutoj, dhe debageri e teston softuerin për të zgjidhur ndonjë problem ose gabim.



# Mundohuni!

Mendoni një situatë problemore në obligimet e përditshme të shkollës. Mundohuni të mendoni dhe të funksiononi si një mjedis i integruar për programim: krijoni hapa zgjidhjeje, kompajloni ato hapa që miqtë tuaj t'i kuptojnë dhe bisedoni me ta rreth elementeve shtesë të mundshëm, përpiquni të zbuloni nëse është zgjidhja e duhur.

Mjedisi i integruar për programim, që do të përdorim kur krijojmë programe me gjuhën programuese **C++ është Code::Blocks**.

**Code::Blocks** është një mjedis i integruar programimi që mbështet kompajlimin dhe testimin e shumë gjuhëve programuese. Punon me një varg kompajluesish. Është një paketë pa pagesë për zhvillimin e softuerit në gjuhën programuese C me kod të hapur dhe përmban mjetet e mëposhtme: editorin për tekst, programe për përkthimin dhe lidhjen, si dhe programe për zbulimin e gabimeve.

#### 4.2.1 Instalimi i Code::Blocks

Për shkak se Code::Blocks është një program pa pagesë, me burim të hapur, mund ta shkarkojmë (Download) nga faqja vijuese:

https://www.codeblocks.org/downloads/binaries.

Në fajllin e pranuar, i jepet urdhëri që të jetë i ekzekutueshëm, përkatësisht të fillon procesi i instalimit me klikim të dyfishtë në të. Pastaj, paraqitet magjistari (**Wizard**) për vazhdim të procesit të instalimit, siç është paraqitur në figurën vijuese:



Figura 1: Fillimi i procesit për instalimin e Code::Blocks

Me klikim në butonin **Next** qasemi në hapin tjetër të procesit të instalimit, nëse i pranojmë parimet e përdorimit të këtij programi:

6	Code::Block	s Installation		-		×
	License Agreen Please review th	<b>nent</b> le license terms ber	fore installing	) CodeB	locks.	
Press Page Down t	o see the rest of the agree	ment.				
GNU ( Vers	GENERAL PUBLIC LICENSE sion 3, 29 June 2007				'	^
Copyright (C) 2007 Free Software Foundation, Inc. < <u>http://fsf.org/</u> > Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.						
1	Preamble					
The GNU General software and othe	l Public License is a free, cop er kinds of works.	pyleft license for				~
If you accept the t agreement to insta	erms of the agreement, dic Il CodeBlocks.	k I Agree to contin	iue. You mus	t accep	t the	
Nullsoft Install System	rv3.02.1					
		< Back	I Agree		Cance	L

Figura 2: Pranimi i rregullave për instalim të programit

Për të vazhduar procesin, klikoni në **I Agree** dhe vazhdoni në hapin tjetër, që është zgjedhja e komponentëve, që duam të instalohen si paketë e këtij programi:

•	Code::Blocks Installation – 🗆 🗙
	Choose Components Choose which features of CodeBlocks you want to install.
Check the components you w install. Click Next to continue	vant to install and uncheck the components you don't want to
Select the type of install:	Full: All plugins, all tools, just everything
Or, select the optional components you wish to install:	Ochault install     Contrib Plugins     C::B CBP2Make     C::B CC Test     C::B Share Config     C::B Launcher
	Description
Space required: 80.9 MB	Position your mouse over a component to see its description.
Nullsoft Install System v3.02.1 -	
	< Back Next > Cancel

Figura 3: Zgjedhja e komponenteve plotësuese në mjedisin e punës

Me klikim në butonin **Next**, hapet dritarja tjetër përmes së cilës zgjedhim folderin ku do të vendosen fajllat, që janë të nevojshëm për funksionim normal dhe të pandëprerë të programit:

5	Code::B	locks Installation		×
	Choose Ins Choose the	stall Location e folder in which to inst	tall CodeBlocks.	
Setup will install ( and select anoth	CodeBlocks in the followi er folder. Click Install to	ng folder. To install in start the installation.	a different folde	r, dick Browse
Destination Fol	der Files (x86)\CodeBlocks		Bro	wse
Space required: 8 Space available:	80.9 MB 135.2 GB			
	m 12 02 1			
Nullsoft Install Syste	SHI V3,02,1			

Figura 4: Zgjedhja e folderit ku do të vendosen fajllat e programit

Pasi të kemi përfunduar hapat fillestar të instalimit, klikojmë në **Install** dhe fillon transferimi i fajllave programor në folderin destinues:

	<b>Installin</b> Please v	<b>g</b> vait while CodeBlocks	s is being installed.	
xtract: worksp	pace.png			
Extract: splas Extract: splas Extract: splas Extract: splas Extract: splas Extract: splas	sh.png 100% sh_0802.png 100% sh_1005.png sh_1211.png 100% sh_1312.png 100% .png			^
Extract: undo Extract: vfold Extract: work Extract: work	o.png der_open.png space-readonly.png space.png			v
soft Install Sys	tem v3.02.1			
		< Doct	Naut >	Cancel

Figura 5: Instalimi i programit

Pasi të përfundoj ky proces, klikojmë në opsionin Next për të vazhduar në hapin tjetër, në fund shtypet **Finish**, paraqitet dritarja për startim Code::Block:

0	Code::	Blocks Install	ation	×
9	Do you w	ant to run Code	::Blocks now?	
		Yes	No	

Figura 6: Dialogu-dritarja për startimin e Code::Blocks

Nga ky dialog zgjedhim opsionin **Yes**, fillon programi dhe shohim sipërfaqen e punës të mjedisit të integruar për programim **Code::Blocks**.

#### 4.2.2 Instalimi i Code::Blocks

Pas fillimit të programit, paraqitet dritarja e punës e programit Code :: Blocks. Le të hedhim një vështrim në desktop dhe të përshkruajmë funksionalitetin e mjeteve!

5	main.cpp [Proekt] - Code::Blocks 10.05	- 0 ×
File Edit View Search Project	Build Debug woSmith Tools Plugins Settings Help $\longrightarrow 1$	
1 🕒 🖬 🕑 🐛 🥆 🗙 1		
	v v	
😳 🕨 🔖 😳 🖾 🛛 Build target: 🛛	ebug v	
💵 🧐 🕆 🔭 🖑 😡 🗔	<i>L</i> .	
Management X	main.cop X	
I project Synthe least	tindise (serves) utily managata add) ret min()	
	Logs & others	×
	🔥 Codentiliods 🔍 Search results 💊 Build log 🗙 🇳 Build messages 💊 Debugger	
	$\longrightarrow$ 4	
l C:\Users\AndrijanaT\Desktop\Proekt	MinDOWS-1252 Line 1, Column 1 Insert Read/Write	default

Figura 7: Mjedisi i punës për Code::Blocks

Mjedisi i punës i përfshinë elementet në vijim:

- 1. menyja me urdhëra;
- 2. shiriti për menaxhim;
- 3. editori për vendosjen e kodit burimor (Source Code);
- 4. dritarja për shkrimin e mesazheve të gabimit.

Hapi i parë në krijimin e një projekti të ri në mjedisin e integruar për programim Code::Blocks, fillon me klikim në menynë **File** → **New** → **Project**, paraqitet dritarja si në vijim:

		Ne	w from tem	plate			2
Projects	Category: <a< td=""><td>Il categories&gt;</td><td></td><td></td><td></td><td><b>2</b></td><td>Go</td></a<>	Il categories>				<b>2</b>	Go
Build targets Files Custom User templates	ARM Project	AVR Project	Code::Blocks	Console application	1	^	Cancel
	D application	Direct/X project	Dynamic Link Library	Empty project			
	FLTK FLTK project	GLFW GLFW project	GLUT	GTK+ project			
	Irrlicht project	Kernel Mode	Lightfeather	Matlab project		~	View as Large icons
IP: Try right-did	ing an item	Driver	project			~	0

Figura 8: Krijimi i projektit të ri

Nga dritarja zgjedhim **Console Application** dhe klikojmë në butonin Go për të vazhduar në hapin e dytë.



Figura 9: Përshtatja e mjedisit të punës

Me klikim në **Next** paraqitet një dritare nga e cila zgjedhim cilën gjuhë programuese do të përdorim, siç paraqitet në figurën vijuese:

	Console application	×
🐼 Console	Please select the language you want to use.  Please make a selection  C++  1	
	< Back Next > Ca	ncel

Figura 10: Zgjedhja e gjuhës programuese për shkruarje

Me klikim në butonin **Next**, qasemi për përcaktim të emrit të projektit dhe lokacionin ku do të vendoset, përkatësisht të memorohet projekti:

	Console application
🐻 Console	Please select the folder where you want the new project to be created as well as its title.  Project title:
	Folder to create project in:
	C: \Users \AndrijanaT \Desktop \
	Project filename:
	Resulting filename:
	<invalid path=""></invalid>
	4
	< Back Next > Cancel

Figura 11: Ndarja e emrit dhe lokacionit të projektit

Së pari e emërtojmë projektin, pastaj definohet se në cilin folder do të vendoset projekti, gjithashtu i vendosim emër edhe fajllit të projektit që krijohet dhe në fund klikojmë në butonin **Next**. Duke klikuar në butonin **Next** zgjedhim kompajluesin:

	Console application X
🐻 Console	Please select the compiler to use and which configurations you want enabled in your project. Compiler: GNU GCC Compiler
	Create "Debug" configuration: Debug "Debug" options Output dir.: bin\Debug\
	Objects output dir.: obj\Debug\
	Create "Release" configuration: Release  "Release" options  Output dir.: bin\Release\  Objects output dir.: obj\Release\
	2 < Back Finish Cancel

Figura 12: Zgjedhja e kompajlerit

Me klikim në butonin **Finish** fillon procedura për krijimin e kodeve burimore në editorin e mjedisit të integruar për programim.



Në sipërfaqen e punës të kompjuterit tuaj krijoni folder, ku do t'i ruani të gjitha projektet e krijuara.

Së pari, krijoni **një fajll të ri** për të shkruar kodin burimor. Fajlli i ri krijohet përmes menysë **File** → **New** → **File**, fillohet me procedurën hap pas hapi. Hapi i parë është të zgjedhim llojin e fajllit:



Figura 13: Zgjedhja e fajllit të C++

Zgjedhim fajllin **C/C++ Source** dhe klikohet në **Go**. Pastaj, vazhdohet në hapin e dytë, që është selektimi i gjuhës programuese:



Figura 14: Selektimi e gjuhës programuese

Zgjedhim gjuhën programuese **C++** dhe klikojmë në butonin **Next** që të vazhdojmë me hapin tjetër të procedurës:

	C/C++ source	×
C/C++ FILE	Please enter the file's location and name and whether to add it to the active project. Filename with full path: C:\Users\AndrijanaT\Desktop\C++ Proekti\{	
	2 2 2 2 2 2 2 4 d file to active project h build target(s): 2 2 2 2 2 2 2 2 2 2 2 2 2	
	All None	·
	3	
	< Back Finish Cance	1

Figura 15: Zgjedhja e fajllit C++

Në dritaren e paraqitur klikojmë në butonin Browse dhe përmes dritares së hapur caktojmë emrin dhe lokacionin e fajllit të ri të krijuar. Duke klikuar në butonin **Finish** futemi në editorin e kodit burimor (**Source Code**) dhe fillojmë me shkruarjen e programit.

# Mbaj mend!

Mjedisi i integruar i programimit është një paketë softuerike që përmban mjetet e nevojshme për krijimin e një programi. Mjete të tilla janë: editori i kodit burimor, bibliotekat e kodeve, kompajluesit dhe platformat e testimit. Code::Blocks paraqet mjedis të integruar pa pagesë për programim me kod të hapur.

# Pyetje

- 1. Çfarë paraqet një mjedis i integruar i programimit?
- 2. Cili është qëllimi kryesor i një mjedisi të integruar të programimit?
- 3. Çfarë mjetesh përfshinëmjedisi i integruar i programimit?
- 4. Në cilën pjesë të dritares shkruhet kodi burimor?
- 5. Cili është roli i kompajluesit në sipërfaqen e punës për programimin?
- 6. Cili është roli i debagerit?
- 7. Përshkruani dritaren e punës së mjedisit të integruar të programimit Code::Blocks!

#### 4.3 Paraqitja e kodeve programore - të shembullit të gatshëm

Për të filluar me shkruarjen e kodit burimor të programit duhet të njohim **strukturën** dhe **sintaksën** e gjuhës programuese. Për shkak se një gjuhë programuese është një grup rregullash, simbolesh dhe shenja të veçanta (speciale), që përdoren për të krijuar një program, ekzistojnë rregulla për **sintaksën (gramatikën)** dhe **semantikën (logjikën)** që duhet të ndiqen me qëllimin e vetëm, krijimin e një programi që do të funksionoj mirë dhe pandërprerë.

Kodi burimor është shkruar në editorët e kodit burimor. Në figurë është dhënë shembull për kodin burimor:

e_para.cpp 🗙	
1	//krijimi i programimit të thjeshtë
2	<pre>#include <iostream></iostream></pre>
3	using namespace std;
4	
5	<pre>int main ()</pre>
6	₽ {
7	<pre>cout &lt;&lt; "Programimi në C++" &lt;&lt; endl;</pre>
8	return 0;
9	}
10	

Figura 1: Shembulli i kodit burimor

Nga editori i kodit burimor vërejmë se secili rresht (vijë) është shënuar me numër. Pranë numrave ekziston një vijë e gjelbër që na tregon se ky kod është përkthyer. Paraprakisht, para se të realizohet përkthimi kjo vijë është e verdhë. Le të shpjegojmë kuptimin e kodit në secilin rresht!

Numri i rreshtave (vijave)	Kuptimi i kodit burimor
Rreshti 1	Me shenjën // fillohet me shkruarjen e komentit ose jepet përshkrimi për atë që është në vijim.
Rreshti 2	Urdhëri për përfshirjen e bibliotekës iostream
Rreshti 3	using namespace std tregon se do të përdoren elementet standarde nga bibliotekat
Rreshti 4	Rreshti i zbrazët i cili injorohet gjatë ekzekutimit
Rreshti 5	Funksioni kryesor <b>main ()</b>
Rreshti 6	Kllapa e madhe e hapur paraqet fillimin dhe kllapa e madhe e mbyllur paraqet fundin e ekzekutimit të urdhërave nga funksioni kryesor <b>main ()</b>
Rreshti 7	cout është urdhër për shtypjen në ekran << janë operatorët për shtypje <b>"Programimi në C++</b> " është tekst për shtypjen në ekran <b>endi</b> është urdhër për fundin e rreshtit (vijës) ";" paraqet fundin e urdhërit
Rreshti 8	return 0; është porosi deri te sistemi operativ se programi është ekzekutuar në mënyrë të sukseshme
Rreshti 9	Funksioni main ()

Figura 2: Tabela me sqarime për secilin rresht



Pas çdo deklarimi vendoset pikëpresje (;) dhe kjo është shenjë që deklarimi përfundon këtu, përveç në rastet tjera që do të paraqiten gjatë krijimit të programeve.

Gjatë shkruarjes së kodit burimor, programuesit i kushtojnë vëmendje pasqyrës së urdhërave të shkruar, me qëllim që të mos kontribuojnë në kaos dhe konfuzion, kur analizojnë kodin burimor. Për shembull, për përkthyesin, nëse do shkruhen urdhërat në një rresht ose do të organizohen në rreshta të veçantë, nuk do të thotë asgjë, sepse pikëpresja (;) tregon qartë se ky është fundi i urdhërit (komandës).

Përkthimi dhe lidhja në kodin e ekzekutueshëm zhvillohet si një fazë e vetme, duke përdorur menynë **Build** → **Build** ose një kombinim i butonëve të tastierës **Ctrl F9**:



Figura 2: Urdhëri Build

Nëse gjatë kësaj faze ndodhin gabime, programuesi qaset në zgjidhjen e tyre dhe përsëri kryen përkthimin dhe lidhjen e kodit burimor. Porosia se është gjetur gabimi paraqitet në fund të dritares së punës në një konizë të veçantë **Build log**:

Logs & others		×			
Code::Blocks × 🔍 Search results × 📝 Cccc × 🥸 Build log ×	📌 Build messages 🗙 📝 CppCheck/Vera++ 🛛 📝 CppCheck/Vera++ messages 🛛 🗙 Cscope 🗙	S Debugge			
mingw32-g++.exe -Wall -fexceptions -g -c "C:\Users\AndrijanaT\Desktop\C++ Proekti\prva\main.cpp" -o obj\Debug\main.o					
Output file is bin/Debug/prva.exe with size 1.50 MB					
Process terminated with status 0 (0 minute(s), 2 second(s))					
0 error(s), 0 varning(s) (0 minute(s), 2 second(s))					
		~			

Figura 3: Korniza Build log

Programin do ta ekzekutojmë me urdhërin **Build** → **Run** ose me kombinimi e butonëve të tastierës **Ctrl + F10**. Pas këtij procesi, në folderin e krijuar ka tre datoteka, përkatësisht fajlla: datoteka burimore, e objektit dhe ekzekutive. Thirrja e urdhërit **Build** dhe urdhërit **Run** mund të bëhet përmes shiritit me mjete që gjenden menjëherë nën shiritin-e menysë:



Figura 4: Ikona e mjeteve për Build dhe Run

Me klikim në **Run**, fillon ekzekutimi i programit, i cili paraqitet në dritare të veçantë, siç është paraqitur në vijim:



Figura 5: Dritarja për fajllin e ekzekutuar

Nga kjo dritare vërejmë se porosia është shënuar në fillim të dritares, ndërsa nën atë, vet kompjuteri ka gjeneruar porosinë **"Press any key to continue..."**, që do të thotë, nëse klikojmë ndonjë buton në tastierë, kjo dritare e programit do të mbyllet.



Gjatë shkruarjes së programit gjithmonë duhet të kemi kujdes me **sintaksën**, **semantikën** dhe **strukturën e gjuhës programore**. Kodi i burimit shkruhet në editor, ku çdo rresht është shënuar me numër. Kodi i burimit përkthehet dhe lidhet dhe zhvillohet si një fazë. Nëse paraqiten gabime, gabimet paraqiten në **Build log**. Ekzekutimi i programit paraqitet në dritare të veçantë dhe si rezultat, paraqiten tre fajlla: burimor, i objektit dhe ekzekutiv.



- 1. Çfarë është kodi burimor?
- 2. Me çfarë urdhëri përkthehet fajlli burimor?
- 3. Sa fajlla krijohen pas përkthimit të fajllit burimor? Kush janë ata?
- 4. Si ekzekutohet programi?

#### 4.4 Ekzekutimi i programeve –të shembullit të gatshëm

Nga njësia mësimore paraprake e njohëm kuptimin e funksionalitetit të elementeve themelore dhe standarde në programin e thjeshtë, që realizon paraqitjen e porosisë në ekran. Për të njohur edhe elementet tjera do të krijojmë një program në të cilin do të kemi mundësi që në mënyrë interaktive të veprojmë gjatë ekzekutimit të programit, përkatësisht të vendosim të dhëna ose vlera nga tastiera. Në editorin e kodit të burimit, në mjedisin e integruar për programim **Code::Bllock**, do të shkruajmë serinë e urdhërave si në vijim:

mbledhja_e_dy_numrave.cpp	x
1	//mbledhja e dy numrave
2	<pre>#include <iostream></iostream></pre>
3	using namespace std;
4	
5	int main ()
6	☐ {
7	<pre>int nr1, nr2, shuma;</pre>
8	<pre>cout &lt;&lt; "Të llogarisim shumën e dy numrave!" &lt;&lt; endl;</pre>
9	<pre>cout &lt;&lt; "Jepe numrin e parë:" &lt;&lt; endl;</pre>
10	<pre>cin &gt;&gt; nr1;</pre>
11	cout << "Jepe numrin e dytë:" << endl;
12	cin >> nr2;
13	shuma=nr1+nr2;
14	<pre>cout &lt;&lt; "Shuma e dy numrave është:" &lt;&lt; shuma &lt;&lt; endl;</pre>
15	return 0;
16	}
17	

Figura 1: Kodi burimor për llogaritje të shumës

Në rreshtin e parë të programit kemi shënuar **komentin**, përkatësisht **sqarimin** për atë se çfarë programi do të krijojmë dhe çfarë do të llogarit programi. Në rastin tonë do të llogaritim shumën e dy numrave.

Çdo program fillon me hyrje, prej ku përkufizojmë se cilat **biblioteka** do të përfshihen dhe cilat urdhëra do t'i përdorim:

2	<pre>#include <iostream></iostream></pre>
3	using namespace std;

Figura 2: Hyrje në program

Në këtë rast është përfshirë biblioteka **iostream**, dhe kjo do të thotë se do të zbatohen urdhërat standard për futjen e të dhënave dhe paraqitjen e rezultatit në ekran. Pastaj vijon funksioni në të cilin ekzekutohen urdhërat që në të vërtetë e përbëjnë programin:

5	int main ()
6	
7	<pre>int nr1, nr2, shuma;</pre>
8	<pre>cout &lt;&lt; "Të llogarisim shumën e dy numrave!" &lt;&lt; endl;</pre>
9	<pre>cout &lt;&lt; "Jepe numrin e parë:" &lt;&lt; endl;</pre>
10	cin >> nr1;
11	cout << "Jepe numrin e dytë:" << endl;
12	cin >> nr2;
13	<pre>shuma=nr1+nr2;</pre>
14	<pre>cout &lt;&lt; "Shuma e dy numrave është:" &lt;&lt; shuma &lt;&lt; endl;</pre>
15	return 0;
16	}

Figura 3: Funksioni kryesor Main

Meqenëse, në këtë program do të futim të dhëna me tastierë, fillimisht në programin kryesor i përkufizojmë variablat, përkatësisht ndryshoret vlerat e të cilave varen nga futja përmes tastierës. Me **int** tregojmë se vlerat që do të jepen me tastierë do të jenë **numra të plotë (integer)**. Variablave u kemi dhënë emra: **nr1**, **nr2** dhe **shuma**.

Në rreshtin me numër (8) është shënuar urdhëri **cout** me të cilin realizohet paraqitja e porosisë në ekran. Pas urdhërit shkruhen operatorët për shtypje (<<), dhe pastaj porosia e cila duhet të paraqitet në ekran e shkruar në thonjëza ("). Me urdhërin **endl** shënojmë fundin e rreshtit, dhe me shenjën pikëpresje (;) fundin e deklarimit, përkatësisht urdhërit. Të njëjtin funksionalitet dhe kuptim e ka edhe rreshti me numër nëntë (9).

Në rreshtin me numër dhjetë (10), është shënuar urdhëri cin, që shoqërohet me operatorët (>>) ">>" paraqet urdhërin për futjen e të dhënave nga tastiera. Gjithashtu, urdhëri përfundon me shenjën pikëpresje (;). E njëjta procedurë është përsëritur në rreshtat njëmbëdhjetë (11) dhe me numër dymbëdhjetë (12).

Në rreshtin trembëdhjetë (13), është shkruar formula për llogaritjen e shumës së dy numrave të dhënë ose në përkthim tekstual variabla shuma do të merr vlerën e shumës së **nr1** dhe **nr2**. Por, ky rresht nuk paraqitet në ekran, pra vetëm përkthehet. Vlera e vendosur në **shuma**, do të paraqitet në rreshtin vijues nga vargu i urdhërave.

Kështu, në rreshtin katërmbëdhjetë (14) me **cout** dhe operatorët për shtypje (<<) do të shënohet fjalia e cila është shënuar në thonjëza, dhe pas saj vlera e variablit **shuma**. Këtu me endl përfundon urdhëri. Funksionin kryesor e përfundojmë me **return 0**; me çka i tregojmë procesorit se programi është ekzekutuar në mënyrë të suksesshme. Me mbylljen e kllapës së madhe shënojmë fundin e urdhërave në funksionin kryesor **main**.

Pas shkruarjes së kodit burimor në **editor** në mjedisin e integruar për programim, qasemi në kompajlim dhe debagim, do të thotë, me përkthim dhe lidhje të fajllit burimor në fajllin ekzekutiv. Përmes menysë **Build** dhe urdhërin **Build** e fillojmë fazën për përkthim. Nëse nuk paraqiten gabime në **Build log** vazhdojmë me urdhërin **Run** për ekzekutim të fajllit. Në të kundërtën i korrigjojmë gabimet dhe e përsërisim procesin. Fajlli ekzekutiv ekzekutohet në dritaren në vijim:

```
Të llogaritim shumën e dy numrave!
Jepe numrin e parë:
24
Jepe numrin e dytë:
56
Shuma e të dy numrave është: 80
Proces returned 0 <0x0> execution time: 11.656 s
Press any key to continue.
```

Figura 4: Ekzekutimi i programit për llogaritje të shumës
Në këtë ekran në të vërtetë e shikojmë rezultatin e punës tonë, dhe verifikohet nëse është zhvilluar procesi sipas hapave të definuar.

Por, nganjëherë nuk shkon gjithçka sipas planifikimit. Është e mundshme të ndodhin gabime gjatë përkthimit, për të cilat programuesi duhet patjetër të gjenë zgjidhje dhe procesi të vazhdoj mëtutje. Gabimet kompjuterike quhen **Bugs**, dhe nga kjo vinë edhe emri i procesit të gjetjes së përmirësimit të gabimeve - **debagimi (Debug)**. Këtë proces praktikisht do ta ushtrojmë përmes krijimit të kodit burimor.

Në editorin për kod burimor **Code::Block** t'i shkruajmë vargjet vijuese nga urdhërat:

prodhimi_i_dy_numrave	scpp X
1	//prodhimi i dy numrave
2	<pre>#include <iostream></iostream></pre>
3	using namespace std;
4	
5	<pre>int main()</pre>
6	E {
7	<pre>int nr1, nr2, prodhimi;</pre>
8	<pre>cout &lt;&lt; "Të llogaritim prodhimin e dy numrave" &lt;&lt; endl;</pre>
9	<pre>cout &lt;&lt; "Jepe numrin e parë:" &lt;&lt; endl;</pre>
10	<pre>cin &gt;&gt; nr1;</pre>
11	<pre>cout &lt;&lt; "Jepe numrin e dytë:" &lt;&lt; endl;</pre>
12	cin >> nr2;
13	<pre>Prodhimi=nr1*nr2;</pre>
14	<pre>cout &lt;&lt; "Prodhimi i dy numrave është:" &lt;&lt; prodhimi &lt;&lt; endl;</pre>
15	
16	system ("PAUSE");
17	return 0;
18	}
19	

Figura 5: Kodi burimor për llogaritje të prodhimit

Të mundohemi t'i shohim gabimet të cilat i kemi bërë gjatë krijimit të programit. Le t'i theksojmë!

Gjatë procesit të përkthimit të fajllit burimor do të fitojmë rezultatin:



Figura 6: Porosia për gabimet e paraqitura

Në fushën **Build log** vërejmë se ekzistojnë gabime gjatë përkthimit të programit. Me klikim në opsionin **Build message** do t'i shohim gabimet një nga një:

ogs & others				
Code::Blocks	× 🔍 s	earch results 🛛 🕺 Cccc 🗶 🥎 Build log 🖂 🌪 Build messages 🛛 📝 CppCheck/Vera++ 🖂 🥂 CppCheck/		
File	Line	Message		
C:\Users\Andri	10	error: expected ';' before 'cin'		
C:\Users\Andri	13	error: expected ';' before 'proizvod'		
C:\Users\Andri	:\Users\Andri 16 error: 'system' was not declared in this scope			
C:\Users\Andri	sers\Andri 7 warning: unused variable 'bri' [-Wunused-variable]			
		=== Build failed: 3 error(s), 1 warning(s) (0 minute(s), 1 second(s)) ===		

Figura 7: Gabimet e ndara sipas rreshtave

Nga dritarja vërejmë se kemi gabime:

- në rreshtin e dhjetë (10) mungon shenja,,;" para thirrjes së urdhëri cin;
- në rreshtin e trembëdhjetë (13) përsëri mungon shenja ";" menjëherë pas definimit të vlerës "Prodhimi";
- në fund nuk e njeh urdhërin system ("PAUSE"), sepse e njëjta nuk i takon bibliotekës iostream.

Ne i qasemi zgjidhjes së gabimeve, përkatësisht shtojmë shenjën ";" në vendet ku mungon dhe në pjesën hyrëse e shtojmë bibliotekën **cstdlib** me qëllim të mundësohet funksionaliteti i urdhërit system **("PAUSE")**.



Meqë nuk ka më gabime e realizojmë ekzekutimin, në ekran të veçantë do të paraqitet fajlli ekzekutiv dhe rezultati i programit.

Figura 8: Kodi burimor pas korrigjimeve të realizuara

## Mbaj mend!

Urdhërat parapërpunuese (paraprocesorike) fillojnë me shenjën "#". Përmes #include i shënojmë bibliotekat e urdhërave që përdoren në kodin burimor. **Main** është funksioni kryesor që e ka çdo C++ program dhe fillon me kllapën e madhe "{" dhe përfundon me kllapën e madhe të mbyllur "}". Në mes dy kllapave definohen variablat, shkruhen deklarimet si pjesë e hapave të programit, shenja, simbole etj. Gjatë përkthimit dhe lidhjes mund të paraqiten gabime të cilat quhen Bugs, dhe procesi i zbulimit dhe përmirësimit të gabimeve quhet debagim (Debug).



- 1. Si fillon secili program në gjuhën programuese C++?
- 2. Definoni hyrjen në program!
- 3. Si quhen rreshtat që fillojnë me dy vija të pjerrëta (//)?
- 4. Tregoni se cila është biblioteka standarde!
- 5. Si quhet funksioni kryesor? Çfarë përmban?
- 6. Si quhen gabimet e programit? Si largohen ato?
- 7. Çfarë do të thotë return 0;?
- 8. Me çfarë shenje ndahen urdhërat nga njëri-tjetri?

## 4.5 Elementet themelore të gjuhës programuese C++

### Të rikujtojmë!

Me komunikimin tonë të përditshëm ne përdorim një gjuhë, dmth këmbejmë informacione, të dhëna, ide, pikëpamje. Si do të kuptohemi në mes vete? Çfarë gjuhe përdorim? Nga çfarë përbëhet gjuha që e flasim? Si i kombinojmë shkronjat për të konvertuar?

Këto pyetje lindin gjithashtu kur bëni gjëra të thjeshta me kompjuterin. Kompjuterët gjithashtu komunikojnë: përdorues - kompjuter - përdorues ose kompjuter - kompjuter. Përdoruesi komunikon me kompjuterin përmes aplikimit të pajisjeve hyrëse përmes të cilave fut të dhëna dhe informacione, jep komanda, përdor programe, etj. Këto aktivitete përkthehen në **gjuhë binare** që është e kuptueshme për kompjuterin, dhe anasjelltas, kur kompjuteri duhet t'i përgjigjet një aktiviteti atëherë shprehjet binare përkthehen në gjuhë të kuptueshme për përdoruesin dhe kështu e shohim rezultatin në ekran.

Komunikimi në mes kompjuterëve realizohet kur janë të lidhur në rrjet dhe mund të këmbejnë të dhëna, informacione, dokumente, programe, pajisje të përbashkëta dhe të ngjashme.

Gjuhët programuese, si gjuhët natyrore, kanë alfabetin e tyre. **Alfabeti i gjuhëve programuese** përbëhet nga:

- shkronjat e mëdha dhe të vogla të alfabetit;
- shifra nga zero (0) deri nëntë (9);
- shenja të veçanta;
- shenja për vend të zbrazët;
- kombinime të dy ose tre shenjave.

Kombinimi dhe aplikimi i duhur i këtyre elementeve krijon **deklarime**, përkatësisht urdhëra dhe si të tilla quhen **elemente ndërtuese të gjuhës programuese**. Prandaj, blloqet ndërtuese të gjuhës programuese janë:

- fjalët e rezervuara;
- identifikatorët;
- operatorët;
- shenjat e interpunkcionit;
- komentet.

Secila prej këtyre blloqeve ndërtuese ka vendin e vet në strukturën dhe sintaksën e gjuhës programuese. Le t'i shpjegojmë ato në mënyrë përshkruese! Fjalët e rezervuara quhen gjithashtu fjalë kyçe. Ato janë paraprakisht të definuara dhe kur përkthejnë fajllin burimor, përkthyesi e di saktësisht kuptimin. Të tilla janë për shembull: int, delete, if, then, else, true, false, while etj. **Identifikatorët** në gjuhën programuese pëkufiz ohen nga programuesi. Ato janë kombinime të karaktereve që mund të kombinohen duke ndjekur rregullat e mëposhtme:

- emri fillon me një shkronjë ose vijë të poshtme;
- shkronjat e mëdha dhe të vogla ndryshojnë, për shembull: "Shuma" është ndryshe nga "shuma";
- emri mund të përmbajë një shifër, por jo të fillojë me shifër;
- emri nuk mund të jetë një fjalë e rezervuar;
- emri nuk mund të përmbajë shenjë speciale si !, @, # etj.

**Operatorët** në gjuhët e programuese përdoren për të treguar aritmetikën, logjikën dhe operacionet e tjera të kryera në program. Shenjat e interpunkcionit përdoren për të ndarë elementet e gjuhës programuese, siç është shenja ";" dhe tregon fundin e një deklarimi, përkatësisht urdhërin (komandën), që do të thotë se deklarimet janë të ndara.

Programuesi mund të shkruaj një koment ose më shumë **komente** gjatë **krijimit të programit**. Shkruarja e komentit fillon me **dy vija të pjerrëta "//"**. Në fakt, komentet përshkruajnë se për çfarë është i dedikuar programi ose shpjegojnë rreshtat individual të deklarimeve ose komandave. Komentet nuk ekzekutohen, ato thjesht qëndrojnë si udhëzues ose shpjegim për kuptimin e elementit, rreshtit të komandës ose programit në tërësi.

Gjuhët e ndryshme programuese ndryshojnë nga njëra-tjetra më shpesh në sintaksë, pra kanë fjalë të ndryshme të rezervuara, rregulla për përdorimin e tyre, etj.

Për shkak se ne punojmë me gjuhën e programuese C++ blloqet ndërtimore saj janë:

- të dhënat me gjuhët programuese mund të përpunohen tipe të ndryshme të të dhënave dhe të gjitha të dhënat kanë këto karakteristika si: emrin, tipin dhe vlerën. Shembuj të tipeve të ndryshme të të dhënave: char, int, bool, float, double, etj;
- 2. konstantet janë madhësi që nuk e ndryshojnë vlerën e tyre;
- 3. ndryshoret janë madhësi që e ndryshojnë vlerën e tyre;
- 4. **deklaratat** në të vërtetë japin një përshkrim të ndryshores, se çfarë tipi i të dhënës do të jetë ndryshorja, për shembull: **int x**, **float a**, etj;
- 5. **shprehjet** për kryerjen e ndonjë llogaritje në varësi të detyrës 110 e cila duhet të. Për shembull: sum + i etj.;

- 6. **deklarimet** (statements) janë urdhëra që duhet të ekzekutohen në program, siç janë: **sum = sum + i, while, for, if-else, switch, break,** etj;
- funksionet në C++ funksioni kryesor është main "()". Deklarimet, pra, urdhërat shkruhen ndërmjet dy kllapave të mëdha të funksionit kryesor, ku çdo deklarim mbaron me shenjën ";". Ekzistojnë funksione që mund të krijohen nga programuesi, por më shumë përdoren funksionet ekzistuese të gatshme;
- 8. **modulet** kur shkruani kodin burimor të programit, secili element shfaqet me një ngjyrë të ndryshme. Përmes ngjyrës e verifikojmë, se a e kemi shkruar saktë fjalën ose deklarimin.

Mënyra në të cilën kombinohen blloqet ndërtuese të gjuhës programuese, rrjedha e tyre dhe radhitja e definojnë **strukturën e gjuhës programuese**. Për shembull, struktura e një programi duket kështu:



Figura 1: Struktura në C++

### Mbaj mend!

Blloqet ndërtuese të gjuhëve programuese janë: fjalët e rezervuara, identifikatorët (identifikuesit), operatorët, shenjat e interpunkcionit, komentet. Gjuha programuese C++ karakterizohet nga këto elemente themelore: të dhëna, konstante, deklarata, ndryshore, shprehje, deklarimet (statetements), funksione dhe module.



- 1. Numëro blloqet ndërtuese të gjuhës programuese C++!
- 2. Numëro elementet themelore të gjuhës programuese C++!
- 3. Çfarë janë fjalët e rezervuara? Numëro disa prej tyre!
- 4. Çfarë janë identifikatorët? Si krijohen ato?
- 5. Jepni disa shembuj, të identifikatorëve që janë krijuar me saktësi sipas rregullave për krijimin e tyre!
- 6. Çfarë janë funksionet? Çfarë funksioni duhet patjetër të ketë programi?
- 7. Çfarë janë komentet? Jepni një shembull për një koment!

### 4.6 Deklarimet

Deklarimet (statements) paraqesin elementet themelore të çdo gjuhe programuese. Nëse nuk ka deklarime, kompjuteri nuk e di se cilat hapa duhen të realizohen për të arritur deri te rezultati përfundimtar. Prandaj, **deklarimet** (mund të përcaktohen si komanda që i tregojnë kompjuterit se çfarë të bëjë, ose çfarë veprimesh duhet të ndërmerr për të arritur një zgjidhje.

Deklarimet në program ndahen me shenjën; "Shenja" na tregon se ky është fundi i deklarimit.

Në gjuhët programuese ka shumë deklarime dhe të gjitha kanë kuptime, funksione dhe qëllime të ndryshme. Kur programojmë shpesh hasim në deklarimet për paraqitjen në ekran dhe deklarimet për ndarje të vlerave.

#### 4.6.1 Deklarimi për paraqitjen në ekran

Gjatë shqyrtimit të programeve të gatshme, programi i parë që takuam ishte programi i thjeshtë që shkruan tekst ose një mesazh në ekran.



Pas përkthimit dhe lidhjes së këtij fajlli (skedari) burimor, në dritaren e ekzekutimit shfaqet mesazhi "Programimi në C++". Sipas shembullit:

cout << "Programimi në C++" << endl;</pre>

Figura 1: Deklarimi për paraqitje në ekran

deklarimi mundëson shfaqjen në ekran. Le të shqyrtojmë elementet që i përmban ky deklarim:

Elementi	Kuptimi
cout	Shprehja për paraqitje në ekran
<<	Operatori për shkruarjen e porosive në ekran
un	Gjithëçka që është shkruar në thonjëza paraqitet në ekran
Programimi në C++	Porosia në mes dy thonjëzave që do të paraqitet në ekran
endl;	Përfundimi i vijës (rreshtit)
;	Përundim i deklarimit

Tabela 1: Elementet e deklarimit të paraqitura në ekran

Termin **cout** mund ta përdorim edhe në mënyra të tjera. Le të shohim se çfarë do të shfaq deklarimi **cout** në ekran, nëse paraqitet në mënyrat si në vijim:

Deklarimi cout	Paraqitja në ekran
cout << "Numri a"	Në ekran do të paraqitet porosia e cila gjendet në mes të thonjëzave
cout << nr	Në ekran do të paraqitet vlera e ndryshores nr
cout << 55	Në ekran do të shtypet numri 55
cout << 3 * nr	Në ekran do të paraqitet rezultati i llogaritjes

Tabela 2: Mënyra të ndryshme të përdorimit të deklarimit cout

Nëpërmjet mjedisit (rrethinës) të integruar të programimit **Code::Blocks** të krijojmë programin, i cili në ekran do të paraqes porosinë, si në kodin burimor të shembullin në vijim:

//Programi për paraqitjen në ekran
<pre>#include <iostream></iostream></pre>
<pre>#include <cstdlib></cstdlib></pre>
using namespace std;
int main ()
cout << "Shokë!" ;
<pre>cout &lt;&lt; "Sot pas informatikës" ;</pre>
cout << "do të mësojmë";
<pre>cout &lt;&lt; "Programimin në C++" ;</pre>
<pre>system ("pause");</pre>
return 0;
}

Figura 2: Kodi burimor për paraqitjen e tekstit në ekran

Rezultati i programit është paraqitur në figurën vijuese:



Figura 3: Mesazhi (porosia) i paraqitur në ekran

Nga figura, vërejmë se mesazhi është shkruar në një rresht, të gjitha pjesët bashkohen dhe kështu të paraqitur nuk tregon dukshmëri dhe qartësi. Për t'i ndarë në rreshta të veçantë do të përdorim komandën **endl**:



Shokë! Sot pas informatikës do të mësojmë Programimin në C++ Press any key to continue...

Figura 4: Aplikimi i urdhërit endl

Në kodin burimor të këtij shembulli, vërejmë se ka shumë deklarime për të shtypur dhe ato do të ekzekutohen një nga një, siç është theksuar. Deklarimet e shkruara në këtë mënyrë quhen sekuenca ose vargje të deklarimeve.

#### 4.6.2 Deklarimipër ndarjen e vlerave

Deklarimin për ndarjen e vlerave e aplikuam gjatë llogaritjes së shumës së dy numrave të dhënë. Ndarja e vlerës së ndryshores mund të realizohet me anë të tastierës ose, si vlerë që fitohet nga llogaritja.

Le të shohim kodin burimor të programit për llogaritjen e shumës së dy numrave!



Nga figura, vërejmë se programi fillon me koment si përshkrim të programit. Pas komentit vijojnë ngjarjet paraprocesorike dhe deklarimet se do të përdoren deklarime standarde që gjenden në **namespace**. Pastaj, fillon funksioni kryesor **main()**.

Figura 5: Ndarja e vlerës së ndryshores

Në funksionin kryesor është paraqitur sekuenca e deklarimeve që do të ekzekutohen në sipas radhitjes. Përveç deklarimit të ndryshoreve dhe deklarimit për paraqitje në ekran, këtu përdoret edhe deklarimipër përcaktimin e vlerës. Procedura për aplikimin e deklarimit **cin**, si deklarim që i jep vlerë ndryshores përmes tastierës, dhe kjo paraqitet në figurën vijuese:

#### cout << "Jepe numrin e parë" << endl; cin >> nr1;

Figura 6: Ndarja e vlerës nga tastiera

Prandaj, **cin** është deklarim përmes së cilës do të caktojmë vlerën e variablës përmes tastierës, përkatësisht do të jepen të dhëna përmes tastierës. Pastaj vijnë operatorët për **vlerën e hyrjes ">>"** dhe ndryshorja e definuar për numrin e parë, përkatësisht **nrl**.

Ndryshorja **nrl** do të ketë një vlerë të dhënë nga përdoruesi përmes tastierës. Por së pari, para se të zbatohet kjo shprehje, duhet të definohet ndryshorja **nrl**. Në figurën e mëparshme, variablat **nrl**, **nr2** dhe **shuma** definohen si vlera **integer**, pra numra të plotë. Prandaj, për të caktuar vlerat **nrl** dhe **nr2** nga tastiera, përdoruesi do të vendos numra të plotë. Kjo paraqet **teknikën për vendosjen e të dhënave**. Ndryshores **shuma** nuk do t'i caktohet vlerë nga tastiera, por vlera e saj do të jetë rezultat i llogaritjes së shumës së dy numrave, siç tregohet në figurën vijuese:

> shuma=n1+n2; cout << "Shuma e dy numrave është:" << shuma << endl; Figura 7: Ndarja e vlerës së ndryshores si rezultat i llogaritjes

Nga paraqitja, vërtetojmë se vlera e ndryshores **shuma** varet nga vlerat e **nr1** dhe **nr2** që përdoruesi do t'i vendos nga tastiera. Shuma e dy numrave përcakton vlerën e ndryshores **shuma**. Në fund, rezultati shfaqet në ekran me deklarimin **cout**.



#### Mbaj mend!

Deklarimet janë komanda që përcaktojnë se çfarë veprimesh duhet të bëjë një kompjuter për të arritur një zgjidhje. Pikëpresja (;) pas secilit deklarim tregon fundin e deklarimit. Deklarimicout përdoret për paraqitjenë ekran. Ndiqet nga operatorët (<<), thonjëzat për shkruarjen e mesazhit dhe përfundon me pikëpresje. Deklarimi për caktimin e vlerës së ndryshores nga tastiera, përkatësisht për futjen e të dhënave është cin. Pas deklarimit cin vijojnë operatorët (>>) për vendosjen e të dhënave të ndryshores, dhe mbaron me një pikëpresje. Këto lloj programesh quhen programe interaktive. Por përveç si vlerës e dhënë nga tastiera, ndryshorja mund të marrë vlerë edhe si rezultat i llogaritjes.



- 1. Çfarë është deklarimi?
- 2. Cili deklarim përdoret për paraqitje në ekran?
- 3. Çfarë mundëson komanda endl?
- 4. Kush është deklarimi?
- 5. Si mund të përcaktohet vlera e tastierës?

## 4.7. Përpunimi i programit

Në detyrat e mëposhtme do të krijojmë programe ku shfaqen shumë deklarime në ekran. Deklarimet e shkruara në kodin e programit do të ekzekutohen një nga një sipas radhës dhe kjo mënyrë e ekzekutimit quhet **teknikë e ekzekutimit me renditje** (radhitje).

Për shembull, me ndihmën e deklarimit për paraqitje në ekran, mund të shtypim formën e trekëndëshit kënddrejtë. Këtu do të përdorim teknikën e ekzekutimit me renditje (radhitje). Për këtë qëllim, krijojmë një projekt të ri në Code::Blocks dhe fillojmë të shkruajmë programin:



Figura 1: Paraqitja e deklarimeve të radhitura për paraqitje në ekran

Rezultati i kodit burimor është paraqitur në figurën vijuese:

*		
**		
***		
****		
****		
****		
*****		
*****		
Press any key to continue		

Figura 2: Paraqitja e deklarimeve të radhitura për paraqitje në ekran

Le të përpiqemi të krijojmë një katror dhe gjatë shfaqjes së katrorit të jetë e dukshme diagonalja. Për këtë qëllim, përveç yjeve të përdorura, do të përdorim edhe një shenjë tjetër. Kodi burimor i këtij programi do të dukej kështu:



Figura 3: Kodi burimor për shfaqjen e katrorit

Rezultati i ekzekutimit të programit është shfaqur në ekran, si në vijim:

*@@@@@@@@@@@@
***@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
****@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
* * * * * @@@@@@@@@@
* * * * * * @@@@@@@
*******@@@@@@
***********@@@
**********@
Press any key to continue

Figura 4: Paraqitja e fajllit ekzekutiv për shfaqjen e katrorit

Në bazë të parimit të njëjtë, të përpiqemi të bëhemi kreativ dhe të vizatojmë objekte me ndihmën e deklarimeve për shfaqje në ekran. Për shembull, mund të vizatoni shkronjën e parë të emrit tuaj, me objekt-zemër, etj.



Shkruarja e deklarimeve në kodin programor, ku të gjitha ekzekutohen në renditje, quhet teknikë e ekzekutimit me renditje.

#### 4.8 Operacionet matematikore dhe shprehjet

Gjatë krijimit të programit në gjuhën programuese C++ më së shpeshti janë përdorur operacionet matematikore dhe shprehjet. Operacionet matematikore dhe shprehjet në të vërtetë mundësojnë llogaritjet matematikore. Prandaj, **shprehjet aritmetike në gjuhën programuese C++ mund të përkufizohen si një rekord i dy ose më shumë vlerave numerike të shoqëruara me operatorë matematikor**.

Operatorët matematikor janë simbole që paraqesin veprim specifik. Operatorët aritmetik bazë që përdoren në C++ janë:

Elementi	Kuptimi
+	Mbledhje
-	Zbritje
*	Shumëzim
/	Pjesëtim
%	Mbetja e pjesëtimit

Tabela 1: Operacionet matematikore në C++



Në gjuhën programuese C++ nuk ekziston operator për eksponent. Por, ekziston një funksion i integruar, që është i definuar në bibliotekën cmath.

Për të parë se si zbatohen operatorët aritmetikë, do të krijojmë një program të thjeshtë që do të kryejë operacione të thjeshta matematikore.

```
1
       //Operacionet aritmetike
 2
        #include <iostream>
 3
       #include <cstdlib>
 4
       using namespace std;
 5
 6
       int main ()
 7
     8
            int numril, numri2, shuma, ndryshimi, prodhimi, herësi;
           cout << "Të llogaritet shuma, ndryshimi, prodhimi dhe herësi i dy numrave!" << endl;
cout << "Jepe numrin e parë:" << endl;</pre>
 9
10
11
            cin >> numril;
12
           cout << "Jepe numrin e dytë:" << endl;</pre>
           cin >> numri2;
13
14
           shuma=numri1+numri2;
15
           cout << "Shuma e dy numrave është:" << shuma << endl;</pre>
16
           ndryshimi=numri1-numri2;
17
           cout << "Ndryshimi i dy numrave është:" << ndryshimi << endl;</pre>
18
           prodhimi=numri1*numri2;
19
            cout << "Prodhimi i dy numrave është:" << prodhimi << endl;</pre>
20
           herësi=numri1/numri2;
           cout << "Herësi i dy numrave është:" << herësi << endl;</pre>
21
            system ("pause");
22
23
            return 0;
24
```

Figura 1: Kodi burimor me zbatim të operacioneve aritmetike

Pjesa hyrëse e programit, si në shembuj të tjerë të programeve, përfshinë bibliotekat dhe komandat e definuara në **namespace**. **Funksioni kryesor main()** është në të vërtetë programi kryesor që përmban deklarimet që do të ekzekutohen për të arritur zgjidhjen.

Siç vërejmë, programi kryesor përcakton variablat ose ndryshoret të cilave do tu caktohen vlera. Vlerat që do tu caktohen atyre janë **numra të plotë (integer)**. Variablat **numril** dhe **numri2** do të kenë vlerë që përdoruesi i jep përmes tastierës, ndërsa variablat **shuma**, **ndryshimi**, **prodhimi** dhe **herësi** do të fitojnë vlerën si rezultat i llogaritjes. Gjatë llogaritjes përdoren operacionet themelore aritmetike. Rezultatin e llogaritjes e shohim në dritaren e ekzekutimit:

```
Të llogaritet shuma, ndryshimi, prodhimi dhe herësi i dy numrave?
Jepni numrin e parë:
897
Jepni numrin e dytë:
760
Shuma e dy numrave është: 1657
Ndryshimi i dy numrave është: 137
Prodhimi i dy numrave është : 681720
Herësi i dy numrave është: 1
Press any key to continue...
```

Figura 2: Operacionet aritmetikore me dy numra



#### Mbaj mend!

Operacionet aritmetike janë operacione të dy ose më shumë vlerave numerike që shoqërohen me operatorë matematikor për të realizuar llogaritje. Në gjuhën programuese C++ përdoren operacionet si në vijim: (+) për mbledhje, (-) për zbritje, (\*) për shumëzim dhe (/) për pjesëtim. Me kombinimin e ndryshoreve dhe operatorëve krijohen shprehjet.



### 4.9. Konstantet dhe ndryshoret

Konstantet dhe ndryshoret janë elemente të gjuhëve programuese. Përdorimi i konstanteve dhe ndryshoreve në krijimin e programeve kërkon respektimin e rregullave për aplikim të drejtë të tyre. Le të njihemi me këto!

#### 4.9.1 Konstantet

**Konstantet** janë të dhëna, të cilëve nuk u ndryshon vlera gjatë kohës së ekzekutimit të programit. Konstanta në gjuhën programuese C++ krijohet në dy mënyra. Një mënyrë është përdorimi i urdhërit (komandës) **#define** dhe mënyra e dytë është vendosja e fjalës **const** para tipit dhe emrit të konstantës. Për shembull:



Figura 1: Definimi i konstanteve

Konstanta e parë që kemi përcaktuar në program është konstanta **pi** e cila ka vlerën **3.141**. Atë e definuam me përdorim të urdhërit **#define** dhe vendoset para se të filloj funksioni kryesor **main()**. Meqenëse kjo komandë vendoset para fillimit të funksionit kryesor **main()**, konsiderohet se ekzekutohet në fazën paraprocesorike dhe për këtë arsye nuk përmban pikëprejse në fund, që në të vërtetë tregon fundin e deklarimit.

Konstanta e dytë është **n**. Kjo përcaktohet në funksionin kryesor **main()**. Në fakt, përveç emrit **n** dhe tipit **int**, përdoret fjala **const**, pra **const n=5**, siç tregohet në program. Në disa raste, nëse nuk përcaktohet tipi i konstantës, atëherë supozohet se është e tipit **int**. Rezultati i llogaritjes është në të vërtetë **5 \* 3.141 = 15.705**, që shfaqet në ekran përmes deklarimit **cout**. Mund të përfundojmë se konstantet ndryshojnë nga njëra-tjetra sipas tipit të të dhënës me të cilën shprehet vlera e saj.

#### 4.9.2 Ndryshoret

Në studimin e matematikës, fizikës, kimisë dhe shkencave të tjera të ngjashme, shpesh hasim me shumë ndryshore (variabla) që kanë ndonjë kuptim. Për shembull: perimetri shënohet me L, sipërfaqja shënohet me S, rrezja me r, diagonalja me d, temperatura me T, shpejtësia me v, etj. Ndryshoret janë të dhëna, vlera e të cilave mund të ndryshohet gjatë ekzekutimit të programit. Ato deklarohen me tipin dhe emrin, ose me tipin, emrin dhe vlerën. Për shembull:

int I, j; int m = 15; float e = 2,78; int a = 5, b = 10

#### 4.9.2.1 Tipi i ndryshores

Çdo ndryshore që do të përdoret në program duhet të definohet, përkatësisht të deklarohet pra, të definohet tipi i ndryshores. Tabela e mëposhtme tregon tipet më të përdorura të ndryshoreve dhe kuptimin e tyre:

Tipe të ndryshoreve	Kuptimi
Char	Ndryshore e karakterit, e cila mund të ketë shenjë-vlerë ose numër të plotë
Int	Ndryshore e numrit të plotë që mund të merr vlera2, -1, 0, 1, 2
Bool	Ndryshore logjike e cila mund të ketë një nga këto dy vlera, e sakt (true) dhe e pasakt (false)
Float	Ndryshore reale me precizitet të thjeshtë
Double	Ndryshore reale me precizitet të dyfishtë

Tabela 1: Tipe të ndryshoreve

#### 4.9.2.2 Vendosja e vlerës në ndryshore

Përveç tipit dhe emrit, ndryshores mund t'i caktohet një vlerë me ndihmën e operatorit "=", siç tregohet në shembujt vijues: **c=8**, **a=10**, **a=b\*3**, **x=x+6**, **x=d=f=2** etj. Kjo procedurë quhet **inicializimi** ose **shtimi i një vlere fillestare**. Kjo do të thotë që ndryshores **c** do t'i caktohet një vlerë tetë (8), ndryshorja a do të merr një vlerë që rezulton nga llogaritja e produktit **b\*3**, ndryshorja **x** do të merr një rezultat nga llogaritja e shumës së **x+6**, ndryshorja **f** do të merr vlerë dy (2), **d** do të ketë vlerën e **f**, ndërsa **x** do të ketë vlerën e **d**. Për shembull:



Figura 2: Inicializimi dhe deklarimi i ndryshoreve

Në këtë program në funksionin kryesor, **rreshti me numër nëntë (9)** përcakton dhe inicializon ndryshoret **a** dhe **b**: **int a=6, b=18**. Në **rreshtin me numër dhjetë (10)** është definuar ndryshorja **shuma**, por nuk është e inicializuar. Kjo do të merr një vlerë pas kryerjes së llogaritjes: **shuma=a+b**, dhe rezultati do të shfaqet në ekran.

## Mbaj mend!

Ndryshoret dhe konstantet kur përdoren në program duhet të definohen, pra të deklarohen, që do të thotë të përcaktohet tipi dhe emri i tyre. Ndryshoret fitojnë vlerë me ndihmën e operatorit bararazi (=). Përcaktimi i vlerës fillestare për ndryshoren quhet inicializim.



- 1. Çfarë është konstanta dhe çfarë është ndryshorja? Cili është ndryshimi në mes tyre?
- 2. Çfarë është deklarata? A mund të deklarohen më shumë ndryshore me një deklarim? jepni një shembull!
- 3. Çfarë është inicializimi? Jepni një shembull!

### 4.10 Deklarimet (teknike) për vendosjen e të dhënave në program

Për të kuptuar esencën e teknikës për vendosjen e të dhënave në program do të krijojmë disa programe.



#### Detyra 1

Të krijohet programi për llogaritjen e katrorit të numrit të dhënë!



Figura 1: Llogaritja e katrorit të numrit të dhënë

Kur krijojmë programin për llogaritjen e vlerës për katrorin e numrit të dhënë, krahas komenteve dhe urdhërave paraprocesorike, që gjenden në hyrjen e programit në funksionin kryesor kemi përdorur deklarime që shfaqen në ekran si dhe deklarimet e të dhënave që jepen nga tastiera. Në fakt, këtu kemi përdorur teknikën e futjes së të dhënave. Ndryshorja për të cilën duhet të futen të dhënat nga tastiera, fillimisht definohet se çfarë tipi është dhe pastaj aplikohet në mënyrë të përshtatshme. Rezultatin e ekzekutimit e shohim në ekran të veçantë:



Figura 2: Rezultati nga llogaritja e katrorit të një numri



Njehsoni sipërfaqen dhe perimetrin e një rrethi! Këtu bëhet fjalë për formula të gatshme matematikore që duhet t'i përdorim gjatë llogaritjes. Për shembull: sipërfaqja e rrethit llogaritet me formulën P=r\*r\*pi, ndërsa perimetri llogaritet me formulën si në vijim: L=2\*r\*pi.

```
//Llogaritja e sipërfages dhe perimetrit të rrethit
2
     #include <iostream>
3
     #include <cstdlib>
4
     using namespace std;
5
 6
     #define pi 3.141
7
8
     int main ()
9
   10
11
         float rrezja, S, L;
12
13
        cout << "Të llogaritet sipërfaqja dhe perimetri i rrethit" << endl;</pre>
14
        15
        cout << "Jepe vlerën e rrezes:" << endl;</pre>
16
17
        cin >> rrezja;
18
        S==rrezja*rrezja*pi;
        cout << "Sipërfaqja e rrethit është:" << S << endl;</pre>
19
20
        L=2*rrezia*pi;
        cout << "Perimetri i rrethit është:" << L << endl;</pre>
21
22
        system ("pause");
23
        return 0;
24
     L}
```

Figura 3: Llogaritja e sipërfaqes dhe perimetrit të rrethit

Duke filluar nga formulat e gatshme matematikore **pi** ka një vlerë konstante prej **3.141** dhe për këtë arsye në pjesën paraprocesorit ne e përcaktojmë atë si një konstante duke përdorur urdhërin **#define**. Ndryshoret i definuam në funksionin kryesor si **float**, pra numra real që përmbajnë numër të plotë dhe një pjesë decimale, sepse **pi** ka një vlerë decimale dhe rezultati i llogaritjes pritet të jetë një numër decimal. Pastaj, zbatohen deklarimet e shfaqjes në ekran dhe deklarimet e caktimit të vlerës, të cilat ekzekutohen në mënyrë të njëpasnjëshme dhe japin rezultatin përfundimtar në një dritare të veçantë:

Figura 4: Rezultati i llogaritjes së sipërfaqes dhe perimetrit të rrethit

Detyrë

Njehsoni sipërfaqen e një trekëndëshi duke përdorur formulën matematikore: S=(a\*h)/2.

1 //Llogaritja e sipërfaqes së trekëndëshit 2 #include <iostream> 3 #include <cstdlib> 4 using namespace std; 5 6 #define pi 3.141 7 8 int main () 9 10 11 float a, h, S; 12 13 cout << "Të llogaritet sipërfaqja e trekëdëshit" << endl;</pre> 14 15 16 cout << "Jepe madhësinë e brinjës a:" << endl;</pre> 17 cin >> a; 18 cout << "Jepni lartësinë e trekëndëshit:" << S << endl;</pre> 19 cin >> h;20 S=a\*h/2; cout << "Sipërfaqja e trekëndëshit është:" << S << endl;</pre> 21 system ("pause"); 22 23 return 0; 24 - 1

Figura 5: Llogaritja e sipërfaqes së trekëndëshit

Për të krijuar programin dhe të fitojmë zgjidhjen e saktë, përdoren formulat matematikore për llogaritjen e sipërfaqes së trekëndëshit. Definohet tipi i ndryshoreve; me ndihmën e deklarimit për shfaqje në ekran, është shtypur teksti si pjesë hyrëse për detyrën, janë vendosur vlerat e ndryshoreve përmes tastierës dhe është bërë llogaritja duke përdorur formulën. Pastaj, kur ekzekutohet fajlli, fitohet rezultati si në vijim:

**************************************		
**************************************		
Jepni madhësinë e brinjës a: 7		
Jepni lartësinë e trekëndëshit: 4		
Sipërfaqja e trekëndëshit është: 14		
Press any key to continue		

Figura 6: Rezultati i llogaritjes së sipërfaqes së trekëndëshit



Krijoni programin, i cili gjatë dhënies së numrit me tastierë do të shfaq paraardhësin dhe pasardhësin e tij.

#### 4.11 Shprehjet krahasuese

Në jetën tonë të përditshme, shpesh ndodhin gjëra për të cilat duhet të marrim vendime dhe në varësi të atyre vendimeve duhet të vendosim se në cilën mënyrë do të shkojmë më tej. Për shembull, pas mbarimit të shkollës së obliguar, jemi në një udhëkryq ku duhet të vendosim nëse do të vazhdojmë shkollimin në universitet apo të përfundojmë këtu. Këtu kemi dy mundësi. Mundësia e parë është nëse vazhdojmë shkollimin në universitet ka mundësi të mëdha për të zgjeruar njohuritë tona dhe kështu të jemi konkurrues në gjetjen e një pune profesionale dhe të paguar më mirë. Mundësia e dytë është të vendosni të mos vazhdoni arsimimin dhe të qëndroni në njohuritë e fituara deri tani, dhe kështu mundësia për konkurrencë në treg zvogëlohet. Sidoqoftë, këto janë vendime të vështira për jetën, por shembull i mirë si hyrje në atë për të cilën do të flasim në këtë përmbajtje mësimore.

## Shënim!

A keni qenë ndonjëherë në një situatë ku duhet të merrni vendim? Si vendosët? Çfarë ju bëri të mendoni në atë mënyrë? Përshkruaj situatën!

Gjëra të ngjashme ndodhin me kompjuterët. Për shembull: kur ndonjë program vendos nëse kushti është i plotësuar dhe nëse është i plotësuar çfarë veprimi duhet të ndërmerr, dhe anasjelltas, nëse kushti nuk është i plotësuar çka mëtutje?!

Shprehjet e këtilla quhen **shprehje krahasuese** dhe përdoruesi i zgjedh ato kur ekziston një alternativë. Kjo do të thotë se do të ndërmerret ndonjë veprim nëse pohimi është i vërtetë dhe ndërmerret ndonjë veprim tjetër nëse nuk është i vërtetë. Për shembull, kur filluam krijimin e programeve, kemi theksuar që **deklarimet (urdhërat)** ekzekutohen në mënyrë të njëpasnjëshme. Kur aplikoni shprehje të tilla krahasuese, rendi i programit humbet, sepse ekzekutimi i tij varet nga saktësia e pohimit të definuar.

Këto shprehje krahasuese në gjuhën programuese C++ quhen **shprehje logjike** ose **shprehje Boolean-e (Boolean)**. Gjatë krijimit të shprehjeve të këtilla shfrytëzohen operatorët për krahasim:

Operatori në C++	Kuptimi	Operatori matematik
= =	E barabartë me	=
!=	Jo e barabaratë me	¥
>	Më e madhe se	>
<	Më e vogël se	<
>=	Më e madhe ose barabartë me	2
<=	Më e vogël ose e barabartë me	≤

#### 4.11.1 Struktura për përzgjedhje nga dy mundësi

Kur krijoni programe me shprehje krahasuese, më shpesh përdoret **struktura me dy mundësi**. Në varësi të vlerës së shprehjes, në këtë strukturë bëhet një zgjedhje ndërmjet dy mundësive. Shprehja mund të ketë një vlerë të **sakt** ose të **pasakt**, domethënë nëse gjendja e shprehjes është e saktë, atëherë do të ekzekutohet një urdhër, dhe nëse nuk është e saktë, atëherë do të ekzekutohet urdhëri tjetër ose këtu programi do të përfundoj.

Duke u bazuar në atë që u tha mësipër, dhe varësisht nga fakti nëse kushti është i plotësuar ose jo, mund të themi se ekzistojnë dy lloje të degëzimit të shprehjeve: degëzimi i **njëfishtë** dhe i **dyfishtë**.

Degëzimi i njëfishtë zbatohet nëse plotësohet kushti i detyrës, atëherë të ekzekutohet ndonjë urdhër, dhe nëse nuk plotësohet kushti, atëherë programi duhet të përfundoj, përkatësisht "nëse kushti, atëherë urdhëri". Për këtë qëllim zabtohet, deklarimi if... then.

Për shembull: kur përdoruesi do të shënoj vitet, atëhere programi duhet të tregoj se a është i mitur (moshës jo madhore) ose jo.



Pas deklarimit if nuk vendoset shenja pikëpresje nga arsyeja se në këtë deklarim duhet të ekzekutohen urdhërat që vijojnë. Me vendosjen e pikëpresjes, do të shënojmë fundin e deklarimit.

Nga programi kryesor mund të shohim se është definuar ndryshorja **vite** tipi vlera e plotë, pra, **int**. Pas shfaqjes në ekran **"Sa vite keni?"**, përdoruesi e jep të dhënën nga tastiera dhe shtyp **Enter**. Atëherë, bëhet një kontroll nëse e dhëna është më e vogël ose e barabartë me tetëmbëdhjetë (18) dhe nëse ky kusht plotësohet, atëherë në ekran paraqitet mesazhi **"Jeni i moshës jo madhore"** - (i mitur).

Nëse nuk plotësohet kushti, përkatësisht nëse jepet vlerë më e madhe se tetëmbëdhjetë (18), atëherë nuk do të kryhet asnjë veprim.

Degëzimi i dyfishtë, në të vërtetë, do të thotë, nëse ndonjë kusht është i plotësuar, atëherë të ekzekutohet urdhëri i definuar, dhe nëse nuk është i plotësuar, atëherë të ekzekutohet një urdhër tjetër, dmth "nëse kushti, atëherë urdhëri], përndryshe urdhëri2". Për këtë qëllim përdoret deklarimi If... else.

Për shembull, në detyrën që paraprakisht e përpunuam le ta vazhdojmë me definimin edhe të një urdhëri tjetër.

```
1
       //Urdhëri if... else
 2
       #include <iostream>
 3
       #include <cstdlib>
 4
       using namespace std;
 5
       int main ()
 6
 7
     - {
 8
           int vite;
            cout << "Sa vite keni" << endl;</pre>
 9
10
           cin >> vite;
11
12
           if (vite <=18)
                cout << "Jeni i moshës jo madhore (i mitur)" << endl;
13
14
           else
15
                cout << "Jeni i moshës madhore" << endl;</pre>
16
17
            system ("pause");
           return 0;
18
19
```

Figura 2: Deklarimi If... else

Prandaj, gjatë futjes së vlerës për **"vite"**, programi verifikon nëse e dhëna është më e vogël ose e barabartë me tetëmbëdhjetë (18). Nëse kushti plotësohet, pra vlera gjendet në intervalin ndërmjet zeros (0) dhe tetëmbëdhjetë (18), atëherë në ekran do të shfaqet mesazhi: **"Jeni i moshës jo madhore (i mitur)!"**. Nëse kushti nuk plotësohet, atëherë vlera (e dhëna) është më e madhe se numri tetëmbëdhjetë (18), atëherë të ekzekutohet veprimi, përkatësisht në ekran të shënohet porosia **"Jeni i moshës madhore!"**:



Figura 3: Rezultati i llogaritjes së sipërfaqes dhe perimetrit të rrethit

Gjatë përdorimit të deklarimit **If... else**, mund të ndodh që urdhëri të përmban më shumë deklarime. Kjo mënyrë e shkruarjes së urdhërave quhet blloku prej deklarimeve dhe ata vendosen në mes të kllapave të mëdha. Për shembull, të japim bllok-deklarimet në programin të cilin e përpunuam:



Figura 4: Bllok-deklarimet në If... else

Kjo do të thotë që pasi të futni të dhënat për ndryshoren **"vite"** do të bëhet një kontroll nëse e dhëna e futur është më e **vogël** ose **e barabartë** me tetëmbëdhjetë (18). Nëse kushti plotësohet, atëherë në ekran do të shfaqen dy mesazhe njëra pas tjetrës, dhe nëse kushti nuk plotësohet, përsëri në ekran do të shfaqen dy urdhëra tjerë, në varësi të plotësimit të kushtit. Blloku i deklarimeve gjithmonë shkruhet ndërmjet dy kllapave.

#### 4.11.2 Shkruarja e programeve me strukturë për përzgjedhje nga dy mundësi

Le të krijojmë disa programe në të cilat do të vendosim kushtin. Duke futur të dhënat do të verifikojmë nëse kushti është i plotësuar ose jo, dhe në përputhje me përgjigjen do të ekzekutohen urdhërat.



#### Detyrë

Të krijohet programi i cili do të verifikoj, se a është numri çift ose tek.

1	//Numri cift ose tek
2	<pre>#include <iostream></iostream></pre>
3	<pre>#include <cstdlib></cstdlib></pre>
4	using namespace std;
5	
6	int main ()
7	
8	int numri;
9	cout << "**********************************
10	cout << "A është numri cift ose tek?" << endl;
11	cout << "**********************************
12	cout << "Jepni nië numër:" << endl;
13	,
14	cin >> numri;
15	
16	if (numri %2==0)
17	
18	cout << "Numri është cift!" << endl;
19	
20	else
21	
22	cout << "Numri është tek!" << endl;
23	
24	
25	system ("pause");
26	return 0;

Figura 5: Verfiikimi se a është numri çift ose tek

Në këtë program, pasi të futni numrin nga tastiera, bëhet verifikimi se a është numri çift ose tek. Sipas rregullave matematikore **"çdo numër që pjesëtohet me numrin dy (2) dhe ka rezultat numër të plotë dhe është pa mbetje, atëherë ky numër është çift"**. Këtu përdoret **operatori "%" (mod)** i cili tregon mbetjen e pjesëtimit. Numri i dhënë i pjesëtuar me numrin dy, duhet të ketë mbetje të barabartë me zero, atëherë në ekran duhet të shkruhet porosia **"Numri është çift"**. Nëse kushti nuk është i plotësuar, të shkruhet porosia **"Numri është tek!"**. Rezultati i ekzekutimit të programit shfaqet në dritaren si në vijim:

*****
A është numri çift ose tek?
*************
Jepni një numër? 3456
Numri është çift!
Press any key to continue

Figura 6: Rezultati i programit për verifikim të numrit se a ësht çift ose tek

Detyrë

Të krijohet programi në të cilin do të jepen dy numra. Pastaj të krijohet kushti, i cili do të verifikoj se cili numër i dhënë është më i madh.

1	
1	//clil numer esnte me i madn
2	#Include <lostream></lostream>
3	
4	using namespace std;
5	interview ()
0	int main ()
/	
8	int numril, numri2;
9	cout << "maximum cout << endl;
10	<pre>cout &lt;&lt; "Cili numër është më i madh" &lt;&lt; endl;</pre>
11	cout << "**********************************
12	cout << "Shënoni numrin e parë:" << endl;
13	cin >> numril;
14	<pre>cout &lt;&lt; "Shënoni numrin e dytë:" &lt;&lt; endl;</pre>
15	cin >> numri2;
16	
17	if (numril>numri2)
18	
19	<pre>cout &lt;&lt; "Numri i parë është më i madh se i dyti!" &lt;&lt; endl;</pre>
20	- }
21	else
22	
23	<pre>cout &lt;&lt; "Numri i dyti është më i madh se i pari!" &lt;&lt; endl;</pre>
24	}
25	
26	system ("pause");
27	return 0;
28	

Figura 7: Programi që verifikon se cili numër është më i madh

Në program pasi të kemi futur të dy numrat përdorim deklarimin **if... else** përmes të cilit krijojmë kushtin për të kontrolluar, se a është numri i parë më i madh se i dyti. Nëse plotësohet kushti, në ekran shkruhet porosia, që numri i parë është më i madh se i dyti, dhe nëse nuk plotësohet, shkruan porosinë se numri i dytë është më i madh se i pari. Rezultati i verifikimit është si mëposhtë:



Figura 8: Rezultati i verifikimit, se cili numër është më i madh

Të krijohet programi që verifikon, nëse një numër është pozitiv apo negativ.

Detyrë

//Desitiv perstiv ere i barabartă me zere	
//FUZILIV, HEUALIV USE I DAIADAILE ME ZEIU	
<pre>#include <iostream></iostream></pre>	
#include <cstdlib></cstdlib>	
using namespace std;	
int main ()	
int numri;	
cout << "**********************************	ndl
cout << "A është numëri pozitiv, negativ apo i barabartë me zero?" << end	1;
cout << "**********************************	ndl
cout << "Shënoni një numër?" << endl;	
cin >> numri;	
if (numri<0)	
<pre>cout &lt;&lt; "Numri është negativ!" &lt;&lt; endl;</pre>	
else	
if (numri==0)	
<b>cout</b> << "Numri është i barabartë me zero" << endl;	
else	
cout << "Numri është pozitiv" << endl;	
system ("pause");	
return 0;	
	<pre>///Diffic, Hegaliv Ose i Dafabatte me zero finclude <iostream> finclude <iostream> finclude <cstdlib> using namespace std; int main () {     int numri;     cout &lt;&lt; "A është numëri pozitiv, negativ apo i barabartë me zero?" &lt;&lt; end     cout &lt;&lt; "A është numëri pozitiv, negativ apo i barabartë me zero?" &lt;&lt; end     cout &lt;&lt; "Shënoni një numër?" &lt;&lt; endl;     cin &gt;&gt; numri;     if (numri&lt;0)         {             cout &lt;&lt; "Numri është negativ!" &lt;&lt; endl;             fin (numri==0)             {             cout &lt;&lt; "Numri është i barabartë me zero" &lt;&lt; endl;             }             else             {             cout &lt;&lt; "Numri është i barabartë me zero" &lt;&lt; endl;             }             else             {             cout &lt;&lt; "Numri është pozitiv" &lt;&lt; endl;             }             system ("pause");             return 0;             }             /// Counter ("Count")             // Count ("Count")             // Count")             // Count &lt;</cstdlib></iostream></iostream></pre>

Figura 9: Verifikimi me if...else se a është numri pozitiv, negativ ose i barabartë me zero

Nga ky program, mund të vërejmë se pasi kemi dhënë një numër nga tastiera krijojmë kushtin, nëse numri i dhënë është më i vogël se zero, atëherë të paraqitet porosia se ky numër është negativ. Në të kundërtën, nëse kushti nuk është i plotësuar, vazhdohet me verifikimin e kushtit, nëse numri i dhënë është i barabartë me zero dhe atëherë nëse ky kusht është i plotësuar, në ekran të shkruhet se numri nuk është as pozitiv as negativ, në të kundërtën të shkruaj porosinë që numri i dhënë është pozitiv. Vërejmë se këtu është aplikuar **dy herë deklarimi If... else**, që ekzekutohen në varësi të plotësimit të kushtit të mëparshëm. Kjo teknikë quhet **teknikë e degëzimit të deklarimeve**. Rezultatin e llogaritjes e shohim në ekran:



Figura 10: Rezultati i verifikimit se a është numri pozitiv ose negativ



Shprehjet krahasuese përdoren gjatë zgjedhjes së dy mundësive. Këto quhen shprehje logjike ose Boolean-e dhe për krijimin e tyre shfrytëzohen operatorët për krahasim. Struktura e programit që përmban zgjedhje të dy mundësive mund të jetë me një degë dhe me dy degë. Në rastin e parë përdoret shprehja if, ndërsa në rastin e dytë shprehja if ... else. Këto deklarime në strukturën e tyre mund të përmbajnë më shumë deklarime dhe shkruhen në kllapa të mëdha.



- 1. Përcaktoni shprehjet krahasuese!
- 2. Tregoni për operatorët të cilët përdoren për krahasim!
- 3. Si mund të jetë degëzimi në programin, ku mund të bëhet zgjedhja e dy mundësive? Cili është ndryshimi midis tyre?
- 4. Shkruaj sintaksën e If dhe if...else!



## 4.12 Struktura për përsëritje në cikël deri në plotësimin e kushtit

Deri më tani kemi krijuar shumë programe që kanë strukturë të ndryshme, deklarime të ndryshme, qëllime dhe zgjidhje të ndryshme. Kur krijoni programe në gjuhën programuese C++, deklarimet ekzekutohen një nga një sipas radhës. Kjo mënyrë e ekzekutimit të deklarimeve quhet **radhitje fizike** e ekzekutimit.

Kur krijuam programe duke përdorur deklarimin **If... Else**, radha e ekzekutimit të deklarimeve ndryshonte në varësi të përmbushjes së kushtit të definuar. Por, shpesh ka nevojë që disa deklarime të përsëriten, pra të përsëriten disa herë. Kjo përsëritje e deklarimeve paraqet ciklin, dhe këto struktura quhen **struktura ciklike** ose **struktura për përsëritje**. Radha e ekzekutimit të deklarimit në cikël quhet **radhitje logjike**, sepse varet nga plotësimi i kushtit.

Sipas kësaj, cikli ekzekuton një ose më shumë instruksione disa herë, derisa një ose grupi i kushteve të plotësohet. Deklarimet që përdoren më shpesh që ekzekutojnë cikle janë: **while**, **do – while** dhe cikli **for**.

Gjatë zbatimit të strukturave për përsëritje janë të mundshme dy situata dhe atë:

- dihet paraprakisht se sa herë do të përsëritet cikli;
- numri i përsëritjeve varet nga kushti dhe ky numër nuk dihet paraprakisht. Kushti mund të gjendet në fillim të ciklit ose, në fund të ciklit.

Cikli **while** është cikli më i thjeshtë që përfundon me plotësimin e kushtit të dhënë, që do të thotë se instruksioni while teston kushtin në të cilin ekzekutohen një varg i urdhërave (komandave) deri në plotësimin e tij. Sintaksa është dhënë si në vijim:

```
While (kushti)
```

```
{
urdhëri 1;
urdhëri 2;
urdhëri 3;...
```

}

Nga sintaksa e paraqitur, instruksioni **while** është strukturë e kontrollit që quhet **cikël** ose **loop**. Instruksioni, përkatësisht urdhëri që duhet të ekzekutohet çdo herë gjatë kalimit të ciklit quhet **trup i ciklit**. Para fillimit të ekzekutimit të ciklit, së pari verifikohet nëse kushti është plotësuar. Nëse kushti është plotësuar, fillon me ekzekutimin e deklarimeve, dhe nuk fillon nëse kushti nuk është plotësuar. Për shembull:

```
1
      //Shkruarja e numrave pre 1 deri 10
2
      #include <iostream>
3
      #include <cstdlib>
4
      using namespace std;
5
 6
      int main ()
7
    8
           int i=1;
 9
           while (i<=10)
10
               cout << i << "\n";
11
12
               i++;
13
           1
14
15
           system ("pause");
16
           return 0;
17
```

Figura 1: Programi për shkruarjen e vargut të numrave prej një (1) deri dhjetë (10)

Nga programi, vërejmë se në funksionin kryesor ndryshorja **"I"** është definuar me tipin "numër i plotë" dhe inicializohet me vlerën një (1). Duke përdorur strukturën e përsëritjes **while**, paraqiten numrat e vargut nga një (1) deri në dhjetë (10) derisa të plotësohet kushti i dhënë me ciklin **while**. Në ekran do të paraqitet rezultati si në vijim:

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Press any key to continue	

Figura 2: Paraqitja e vargut të numrave prej një (1) deri dhjetë (10)

#### Shënim!

Në këtë program vargu i numrave paraqitet vertikalisht. Për të paraqitur numrat horizontalisht, fshihet shprehja "\ n" dhe në thonjëza shtohet presje.

Programi i ngjashëm që i shënon shenjat në ekran deri në një numër të caktuar, është paraqitur si në vijim:

```
//Shkruarja e vargut prej 10 shenjave
 2
       #include <iostream>
       #include <cstdlib>
 3
 4
       using namespace std;
 5
 6
       int main ()
 7
     ⊟ {
 8
           int i=1;
 9
           while (i<=10)
10
11
               cout << 'A' << ",";
12
               i++;
13
14
15
           system ("pause");
16
           return 0;
17
```

Figura 3: Paraqitja e vargut prej shenjave

Ciklet (unazat) që shfrytëzojnë instruksionin **while** mund të jenë:

- cikle të kontrolluara nga numëruesi;
- cikle të kontrolluara nga ngjarjet.

**Ciklet e kontrolluara nga numëruesi** shfrytëzojnë ndryshoren e cila e kontrollon ciklin. Për këtë qëllim e inicializon ndryshoren e kontrollit të numëruesit dhe për çdo cikël të ekzekutimit, numëruesi rritet. Për shembull:



Figura 4: Cikli i kontrolluar nga numëruesi

Sipas detyrës, fjalia **"Programimi në C++"** do të shkruhet derisa të plotësohet kushti: **të shkruhet pesë herë**. Në secilin cikël (unazë) numëruesi i rritet për një (1).

Te **ciklet e kontrolluara nga ngjarjet**, kushti për ndërprerje varet nga ndonjë ngjarje që ndodh derisa trupi i ciklit ekzekutohet. Për shembull: të llogaritet prodhimi i numrave të dhënë, dhe vendosja e të dhënave do të ndërpritet kur jepet zero (0) nga tastiera:



Figura 5: Cikli i kontrolluar nga ngjarja

Ky program i ciklit While, do të ekzekutohet derisa të jepet zero (0) nga tastiera. Ndërsa kjo do të thotë se vendosja e zeros (0) është ngjarje që do të ndërpres ciklin. Për këtë thuhet se ciklet e tilla janë të kontrolluar nga ngjarjet. Rezultati i ekzekutimit të kodit paraqitet në figurën vijuese:

```
Vendosja e të dhënave dhe llogaritja do të përfundoj me dhënien e 0!
Jepni një numër: 12
Jepni një numër të ri: 8
12
Jepni një numër të ri: 4
96
Jepni një numër të ri:
384
Jepni një numër të ri: 2
1920
Jepni një numër të ri: 1
3840
Jepni një numër të ri: 0
3840
Prodhimi i numrave të dhënë është: 3840
Press any key to continue...
```

Figura 6: Rezultatet e ekzekutimit të ciklit

# 4.12.1 Shkruarja e programeve të thjeshta me strukturë për përsëritjen e ciklit kur plotësohet kushti

Në strukturat me përsëritje deri në plotësim të kushtit, kushti mund të përcaktohet në **fillim të ciklit** ose në **fund të ciklit**. Në rastin e parë, fillimisht kontrollohet nëse kushti është plotësuar, në mënyrë që vargu i deklarimeve të ekzekutohen, dhe në rastin e dytë deklarimet ekzekutohen derisa të plotësohet kushti. Për shembull, do të llogaritim prodhimin e numrave të njëpasnjëshëm deri te numri **n**, në dy mënyra:

Mënyra e parë: kushti ndodhet në fillim të deklarimit për përsëritjen e ciklit

```
//Prodhimi i numrave të njëpasnjëshme deri te numri n
 1
 2
       //Kusht është vendosur në fillim
      #include <iostream>
 3
 4
      #include <cstdlib>
 5
      using namespace std;
 6
 7
       int main ()
    8
 9
           int i. n:
10
           float prodhimi;
11
           cout << "Për sa numra do të llogaritim prodhiminë" << endl;</pre>
12
           cin >> n;
13
           i=1;
           prodhimi=1;
14
15
           while (i<=n)</pre>
16
           {
17
               prodhimi *=i;
18
               ++i;
19
20
           cout << "Prodhimi i" << n << "numrave është e barabart me" << prodhimi;</pre>
21
           system ("pause");
           return 0;
22
23
```

Figura 7: Kushti i ciklit është definuar në fillim

Mënyra e dytë: do të ekzekutohen vargjet nga deklarimet derisa të plotësohet kushti:



Figura 8: Programi në të cilin ekzekutohen deklarimet deri në plotësimin e kushtit

Detyrë

Duke zbatuar strukturat për përsëritje të ciklit derisa të plotësohet kushti, të krijohen programet si në vijim:

- 1. Të shfaqen në ekran numrat çift deri në numrin n!
- 2. Të llogaritet shuma e numrave tek deri në numrin n!
- 3. Të llogaritet prodhimi i numrave çift deri në numrin n!

## Mbaj mend!

Radha e ekzekutimit të deklarimeve mund të jetë fizike dhe logjike. Përsëritja disa herë e deklarimeve paraqet ciklin (unazën), ndërsa strukturat quhen struktura ciklike ose struktura për përsëritje. Më shpesh si deklarime më të përdorura për cikle janë komandat: while, do-while dhe for. Gjatë përdorimit të komandës while fillimisht verfikohet kushti, dhe pastaj ekzekutohen deklarimet. Te komanda do-while ekzekutohen deklarimet derisa të arrihet kushti.

## Pyetje

- 1. Çfarë janë strukturat për përsëritje? Definoni!
- 2. Cilët deklarime janë më të përdorura për ciklet?
- 3. Përshkruani sintaksën për komandën while dhe do-while!
- 4. Si kontrollohen ciklet gjatë zbatimit të while dhe do-while?
- 5. A dihet se sa herë do të përsëritet cikli? Sqaroni!

## TË PËRSËRISIM! TË USHTROJMË!



### Detyra 1

Shkruani programin me të cilin në ekran do të fitoni paraqitjen si në vijim: Ora e parë: INFORMATIKË Programimi në C++



#### Detyra 2

Cilat janë vlerat e ndryshoreve a dhe b, pas ekzekutimit të urdhërave si në vijim: a=15 b=23 a=a+b b=a-b

a=a-b



#### Detyra 3

Çfarë do të paraqitet në ekran pas ekzekutimit të kodit si në vijim:

int a, b,c;

cin >> a;

cin >> b; cin >> c;

cout << a << "," << b << "," << c << ". " << endl;

nëse përdoruesi i shënon vlerat e njëpasnjëshme si në vijim: pesë (5), gjashtë (6) dhe shtatë (7)?



#### Detyra 4

Çfarë do të paraqitet në ekran pas ekzekutimit të vargut të urdhërave si në vijim? int a=8; int b=10; cout << (a+b)/2.



#### Detyra 5

```
Sa herë do të ekzekutohet cikli si në vijim?
int i=2
do
{
cout << i << " " ;
i ++;
}
While (i<=10)
```

#### 138

## Detyra 6

Shkruaj një program për llogaritjen e sipërfaqes së drejtkëndëshit. Dalja e ekranit të paraqes dimensionet e brinjëve dhe rezultatin e llogaritjes së sipërfaqes. Formula për llogaritjen e sipërfaqes së drejtkëndëshit është: S=a\*b.



#### Detyra 7

Shkruaj një program që llogarit mesataren aritmetike të dy numrave. Si do të realizohet llogaritja e mesatares aritmetike të tetë numrave?



#### Detyra 8

Të krijohet programi që verifikon se a është numri i dhënë çift ose tek.



## Detyra 9

Të krijohet programi që llogarit perimetrin në katror, nëse vlera e dhënë për brinjën "a" është pozitive. Në të kundërtën, të shënohet porosia "Llogaritja nuk mund të realizohet! Të jepet një numër pozitiv".



#### Detyra 10

Me zbatim të ciklit do – while, të llogaritet shuma e numrave çift deri te numri n.



### Detyra 11

Të llogaritet shuma e numrave që në mënyrë ciklike jepen nga tastiera, përderisa nuk është dhënë numri zero (0).

## Të jetuarit onlajn

Hyrje në ueb ditarët (blogjet) Koncepti për blogun dhe zbatimi i tij Përpunimi i blogut Komentimi i përmbajtjeve të blogjeve TË PËRSËRISIM! TË USHTROJMË!

. . .

• • •

## 5. Hyrje në ueb-ditarët (blogjet)

## Të rikujtojmë!

Çfarë janë programet? Jepni përkufizime për programimin! Si përpunohet një program? Cilat gjuhë programuese dini?

Shoqëria moderne dhe trendi i globalizimit masiv imponojnë nevojën për aplikimin e **teknologjisë së internetit** në procesin e arsimit, por edhe në jetën e përditshme në përgjithësi. Zbatimi i teknologjisë së internetit lejon përdoruesit të përdorin shërbime të ndryshme, të tilla si: komunikimi, mesazhet, këmbimi i informacionit, blerja përmes internetit, shkarkimi i imazheve, muzikës, përmbajtje video, etj.

Prandaj, **Interneti** është rrjeti më i njohuri në botë i përbërë nga miliona kompjuterë në të gjithë botën, të cilët janë të ndërlidhur në mënyra të ndryshme në mënyrë që të komunikojnë dhe të bëhet këmbimi i informacioneve. Megjithatë, shërbimi më i përdorur në internet është komunikimi, pra këmbimi i të dhënave dhe informacioneve përmes postës elektronike (e-mail), shërbimeve onlajn për komunikim, rrjeteve sociale, por në kohën e fundit edhe aplikimi i **ueb-ditarëve** ose **blogjeve**. Për shkak se këtu bëhet fjalë për komunikim, përdoruesi duhet të respekton rregullat e përdorimit etik të internetit:

- mosndaj të dhëna personale, si dhe të dhëna të anëtarëve të familjes suaj, të tilla si emri dhe mbiemri, adresa, telefoni ose fotografi me bashkëbisedues të panjohur në anën tjetër të kompjuterit. Nuk kanë qëllime të mira të gjithë njerëzit që komunikojnë në internet;
- mos e përdorni kurrë numrin e kartelës kreditore të prindërve tuaj në internet për të blerë ose porositur ndonjë gjë pa lejen e tyre;
- mos i zbuloni fjalëkalimet që i përdorni për të hyrë në shërbimet e internetit, që të mos keqpërdoren;
- kur bisedoni në internet, mos përdorni fjalë fyese dhe të pa hijshme;
- kur komunikoni në internet, respektoni pikëpamjet e njerëzve të tjerë, edhe pse ato mund të jenë të ndryshme nga tuajat. Në komunikim në internet të silleni në mënyrë miqësore;
- kur shkruani tekst në internet, mos shkruani me germa të mëdha, është njësoj si të bërtisni.

Kohët e fundit, me aktualizimin e rrjeteve sociale, shfaqet termi **gjurmë** (shtypje) digjitale, si rezultat i postimeve që përdoruesi publikon në internet. Në surfimin tonë të përditshëm mendojmë se jemi të mbrojtur dhe anonimë, por nuk është kështu. Kjo tregon se aktivitetet e përdoruesve të rrjetit social monitorohen nga shumë njerëz më shumë nga ajo që mendojmë. Përmes aktiviteteve të ndërmarra nga përdoruesi: pëlqimi i faqeve dhe postimeve, ndarja e lidhjeve, statuseve, imazheve, videove, komentimi i përmbajtjes, etj., formohet **gjurma digjitale** (shtypja digjitale) e përdoruesit, pra krijohet bazë e të dhënave e aktiviteteve të secilit përdorues, përmes së cilës është i vetëdijshëm se për çfarë personi bëhet fjalë, cilat janë interesat dhe kuriozitetet e tij dhe si të ndikohet te ai.

# Sipas kësaj, **gjurmët digjitale janë të gjitha informacionet në internet, (pozitive dhe negative) të cilat janë vendosur rastësisht ose me qëllim**.

Gjurmët digjitale i kanë edhe anët pozitive edhe negative, por ato duhet të kontrollohen. Përshembull, ana pozitive e gjurmëve digjitale është kur kërkojmë një recetë në Internet dhe përmes asaj recete mund të arrijmë në shumë faqe të kuzhinës. Por ka edhe anë negative, si rrjedhim i përdorimit të papërgjegjshëm të internetit. Për shembull, fotografitë që menduam se ishin private, u bënë publike, të dhënat personale në kohë të caktuara nuk janë të mbrojtura mirë etj. Prandaj, para se të postohet ndonjë gjë në internet, përdoruesi duhet të mendojë me kujdes për përfitimet e postimit dhe se si mund të jetë edhe i dëmshëm. Përpara çdo postimi përdoruesi duhet të vlerësoj për vlerat pozitive dhe negative për të marrë vendimin e duhur para postimit.

Këshillat më të zakonshme për gjurmën digjitale, janë si dhënë në vijim:

- dilni nga llogaria kur largoheni nga rrjetet sociale ose nga posta elektronike;
- mos i tregoni fjalëkalimet;
- mos shkruani publikisht fjalëkalimet;
- kur krijoni një fjalëkalim, përdorni shkronja, numra dhe karaktere;
- fshini arkivin më shpesh;
- mos i ndani të dhënat personale në internet.

Në këtë temë do të përqendrohemi në **ueb-ditarët**, pra në **blogje**, si mjet i fuqishëm në internet që lejon komunikimin ndërmjet mësuesve dhe nxënësve, ndarjen e njësive mësimore, ushtrime, video, material shtesë për ata që duan të mësojnë më shumë nga ajo që ofrohet nga librat shkollorë etj.

Për të kuptuar funksionalitetin dhe mundësitë e ofruara nga blogjet me ndihmën e ueb kërkuesit (web browser), le të përpiqemi të gjejmë blogje nga autorë të ndryshëm dhe për tema të ndryshme. Ka shumë blogje që kanë përmbajtje posaçërisht për lëndën që mësojmë, **informatikën**.

## Termat kryesorë!

Ueb-ditarët, blogu, Blogeri, logimi, emri shfrytëzuesit, fjalëkalimi, rekordi, komenti, regjistrimi, gjurmë digjitale.
## 5.1 Koncepti për blogun dhe zbatimi i tij

**Blogu** ose **ueb-ditarët** paraqesin ditarët-onlajn, ku përmbajtja paraqitet sipas rendit kronologjik, përkatësisht lajmet më të reja ose përmbajtjet paraqiten të parët, dhe pjesa tjetër shkon më poshtë, së pari fillojnë të shfaqen lajmet ose përmbajtjet e fundit dhe pjesa tjetër shkon më poshtë dhe janë renditur duke kaluar nga postimet më të fundit te postimet e vjetra.

Fjala blog vjen nga fjalët **web** që do të thotë rrjet dhe **log** që ka kuptimin e vendosjes apo futjes.

Në fakt, blogjet janë një mjet komunikimi që lidh përdoruesit, grupet e biznesit dhe ueb-master të tjerë të të njëjtit mendim.

Autori i blogut quhet **bloger**. Ai jo vetëm që krijon blog, por azhurnon dhe editon vazhdimisht përmbajtjen e tij. Krijimi, azhurnimi dhe redaktimi i përmbajtjes së blogut quhet **blogim (Blogging)**. Një bloger mund të jetë kushdo që ka njohuri të teknologjive kompjuterike dhe të Internetit dhe është i gatshëm të diskutoj një temë të caktuar.

Përmbajtja e blogut mund të jetë e ndryshme: tekste, shënime, diskutime, fotografi, video, adresa në lokacione të tjera në internet, etj. Blogerët e dinë që nga fillimi se çfarë duan të shkruajnë, të komentojnë ose të blogojnë. Temat më të zakonshme të blogjeve janë si më poshtë:

- manualet arsimore dhe temat arsimore;
- personalitet të famëshme: biografitë dhe arritjet e tyre;
- sporti;
- shëndeti dhe bukuria;
- tema për prindërit dhe fëmijët; lojra kompjuterike;
- kafshët shtëpiake dhe përkujdesja ndaj tyre;
- receta;
- politika;
- udhëzues për zona të ndryshme;
- përvojat e jetës;
- shërbime dhe produkte;
- udhëtime;
- moda;
- dekorimi i shtëpisë etj.

Sigurisht, ka shumë tema të tjera që janë me interes për përdoruesit. Blogeri vendos për cilën temë të krijoj një blog. Më shpesh, blogerët zgjedhin tema që i njohin, për të cilat ata kanë njohuri paraprake dhe me ndihmën e diskutimeve për të zgjeruar dhe thelluar njohuritë e tyre, janë të destinuara për publikun, ose për një grup specifik të përzgjedhur që diskuton temën e postuar në blog. Diskutimet mund të jenë argëtuese dhe udhëzuese, gjithashtu mund të gjendet edhe përmbajtja që ka vlerë të madhe për popullacionin e internetit.

Për të mësuar në lidhje me mundësitë e blogjeve, do të aftësohemi të krijojmë blog dhe të jemi pjesë e bashkësisë së blogut.



Blogu ose ueb-ditari paraqet lokacionin në Internet në formën e revistës elektronike, në të cilën përmbajtjet shfaqen me renditje kohore të kundërt (kronologji të kundërt). Blogeri është autori i ueb-ditarit ose blogut. Blogimi është procesi i krijimit, redaktimit, komentimit dhe mirëmbajtjes së blogut. Respektimi i rregullave të sjelljes etike gjatë komunikimit në internet vlen edhe për blogjet dhe gjurmën digjitale (shtypjen digjitale). Gjurma digjitale është gjurma që lëmë pas aplikimit të shërbimeve të ndryshme në internet.



- 1. Çfarë është blogu?
- 2. Çfarë janë blogerët?
- 3. Kush mund të jetë autor i blogut?
- 4. Çfarë ëhtë blogimi?
- 5. Çfarë temash mund të përdoren për krijimin e blogut?
- 6. Kush i shkruan përmbajtjet e bllokut?
- 7. Numëroni disa rregulla për përdorimin e duhur të blogimit!
- 8. A mundet që blogjet të zbatohen si mjete për qëllime shkollore? Si?
- 9. Çfarë është gjurma digjitale?



### 5.2 Përpunimi i blogut

Krijimi i blogjeve është proces i thjeshtë. Mënyra më e thjeshtë, më e shpejtë dhe më e lirë për zbatim të mundësive të tij, specialiteteve, dhe funksionaliteteve është përmes blog-shërbimeve pa pagesë. Ekzistojnë shumë blog-shërbime (servise) për përpunim të blogjeve:

- 1. Blogger; 7. Squarespace;
- 2. Drupal; 8. Tumblr;
- 3. Ghost; 9. Typepad;
- 4. Joomla; 10. Weebly;
- 5. Magento; 11. Wix;
- 6. Medium; 12. WordPress.

Blog-shërbime më të popullarizuara janë Blogger dhe WordPress.

Blogger-i është një nga platformat më të vjetra të **Google**, meqenëse është një pjesë e paketit të **Google**, me lehtësi i qasemi duke përdorur llogarinë tonë personale (**account**) në **Google**, përkatësisht **Gmail**-it. Për këtë, për të filluar me blogimin në Blogger, më së miri duhet të kemi emrin e përdoruesit dhe fjalëkalimin tonë (**Gmail account**).



Figura 1: Logoja e Blogger-it

Meqenëse **Blogger**-i është hostim onlajn, krahas tij, përdoruesi fiton edhe nën domen që duket kështu: emriisajtit.blogspot.com.

Mundësitë e **Blogger**-it mund të përdoren për: shkruarjen e ditarëve, vendosjen e fotografive, shkruarjen e punimeve, portale me lajme dhe risi, doracakë etj. Pastaj, nuk është i kufizuar numri i blogjeve të cilët autori mund t'i krijoj dhe t'i mirëmbaj. Krahas kësaj, përdoruesi mund të realizoj edhe profit (fitim) nga blogu përmes reklamimit, promovimit të prodhimeve të ndryshme etj.

### 5.2.1 Hapat për krijimin e blogut

Për të filluar procedurën e përdorimit të krijimit të blogut në shërbimet pa pagesë të Blogerit (Blogger), në fillim duhet të krijohet llogari në Gmail, pastaj të startohet shfletuesi i uebit (Browser), si për shembull: Google Chrome, Internet Explorer, Mozilla Firefox, Opera etj. Në shiritin e adresës së shfletuesit të uebit shënohet adresa. Për të filluar procedurën e përdorimit të krijimit të blog-ut në shërbimet pa pagesë të Blogerit (Blogger), në fillim duhet të krijojmë llogari në Gmail, pastaj të startojmë shfletuesin e uebit (Browser), si për shembull: Google Chrome, Internet Explorer, Mozilla Firefox, Opera etj. Në shiritn e adresës së shfletuesit të uebit e shënojmë adresën e blog-servisit pa



pagesë www.blogger.com, pastaj paraqitet dritarja si në vijim:

Figura 2: Ekrani fillestar gjatë thirrjes së faqes për krijimn e blogut

Nëse përdoruesi nuk ka llogari (**Account**), klikoni në opsionin **Sign in** nga ku fillohet procedura për regjistrim në **Gmail**, përndryshe, nëse ka llogari në Gmail, atëherë e zgjedh opsionin **Create New Blog**, dhe hapet dritarja për paraqitje në Gmail:



Figura 3: Dritarja për paraqitje në Gmail

Pas logimit (kyçjes) të përdoruesit, përdoruesi i ndanë emër blogut, adresë dhe zgjedh temën, përkatësisht pamjen e blogut. Kjo bëhet me plotësimin e fushave nga figura si në vijim:

litle				— Titulli i blogut
Address	E.g. mynewbloga	ddress.blogspot.com		Adresa e blog
		You	u can also add a custom domain I	ater.
Theme		Soho	Taly Teals	_ e blogut
		Soho Soho		
	Man of a constraint of a second of a secon			
	Notable	Simple	Dynamic Views	_

Figura 4: Krijimi i blogut

# Shënim!

Përveç temave, pamjet që janë dhënë si opsion, përdoruesi mund t'i shton nga lista për oferta.

Kur përdoruesi do të shënoj adresën e blogut, atëherë në anën e majtë në afërsi të kornizës për vendosje paraqitet porosia:

**Checking Address Availability**, dhe kjo do të thotë se realizohet verifikimi, nëse adresa e dhënë është e disponueshme, kjo tregon se nuk është shfrytëzuar nga ndonjë përdorues tjetër. Pastaj, zgjedhet tema ose pamja e blogut e cila është adekuate me përmbajtjen e blogut. Në fund, zgjedhet butoni **Create Blog** (*krijo Blog*) dhe sigurohu që të kontrollosh tabelën e kontrollit (**Dashboard**) në blog. Përmes tabelës së kontrollit përdoruesi e rregullon përmbajtjen e blogut, shton përmbajtje të reja, menaxhon me komentet etj.

shtrime praktike ne 🝷		All labols	0
ew blog	New post Using Blogger as ProektBlog	All labels +	Q
Posts All Stats Comments Earnings Pages Layout	European Union laws require you to give European Union visitors information about cookies used and data collected on your blog. In matthese laws also require you to obtain consent. As a courtesy, we have added a natice on your blog to explain Google's use of certain Blogger and Google cookies, including use of Goc and AdSense cookies, and other data collected by Google. You are responsible for confirming this notice actually works for your blog, and that it displays. If you employ other cookies, for example to party features, this notice may not work for you. If you include functionality from other providers there may be extra information collected users. Learmore about this notice and your responsibilities. Publish Revert to drait	ny cases, X wgle Analytics wy adding third from your X	· > 25 *
<sup>1</sup> Theme	There are no posts. Create a new post		

Figura 5: Tabela e kontrollit (Dashboard) në blog

#### 5.2.2 Rregullimi i përmbatjes së blogut

Pas shfaqjes së tabelë së kontrollit (**Dashboard**), përdoruesi mund të fillojë me rregullimin (redaktimin) e përmbajtjes së bllokut. Që të krijohet postimi/ shpallja (**Post**), klikohet në butonin **New post**, dhe paraqitet dritarja si në vijim:

shtrime praktike	. · Post Detyra 1:	Posting as ProektBlog	Publish Save Preview Close	
Compose HTML		<b>I</b> . **	✓ Post settings	
	Në programin për llogaritje tabelare, të krijohet tabela "Radhitja e orarit".		Sector Labels	
	Tabela të jetë e formatuar me fontin Arial, madhësia 12, të zgjidhet ngjyra e kufizimeve (vijave sipas dëshirës. Qeliat e titujve në tabelë të jenë të ngjyrosura që përputhen me ngjyrat e qeliave sipas të balta.	meve (vijave) grat e qeliave	<ul> <li>Schedule</li> <li>Permalink</li> <li>Q Location</li> <li>☆ Options</li> </ul>	
	ijera te tabeles.			

Figura 6: Tabela e kontrollit (Dashboard) në blog

Me klikim në butonin **Publish** bëhet postimi përmbajtjes së blogut. Postimi mund të ruhet me klikim në butonin **Save**, duke e shikuar me klikim në butonin

preview, si opsion për mbylljen e postimit (publikimit), përkatësisht **Close**. Nga ana e majtë e dritares vërjemë se ka opsione për redaktime plotësuese (**Post settings**) të postimit (**Post**), si për shembull: shtimi i etiketave (tags), nëse postimi në mënyrë automatike do të paraqitet me klikim në butonin **Publish**, ose përdoruesi do të vendos datë dhe kohën e postimit, shtimin e lokacionit nga ku është bërë postimi, përkatësisht publikimi etj.

# Shënim!

Kur publikohet postimi, përsëri si ekran kryesor shfaqet Dashboard. Në tabelën e kontrollit (Dashboard) ka qasje vetëm administratori i blogut. Me klikim në linkun Post mund të shihet lista e postimeve.

Nga dritarja e krijimit të postimit vërehet se shkruarja, editimi dhe formatimi i përmbajtjes së postimit, është i njëjtë si në **MS Word**, që do të thotë që përdoruesi mund të zgjedhë shkronjën, madhësinë e shkronjës, ngjyrën e shkronjës, stilet e shkronjave, të shtojë linqe, të shtoj fotografi etj.

#### 5.2.3 Shtimi i figurës (imazhit) dhe videos në blog

Kur dritarja e krijimit të postimit është e hapur, së pari vendoset kursori në vendin ku do të shtohet figura dhe zgjedhet mjeti për shtimin e figurës:

23				
			-	
	-			

Figura 7: Mjeti për shtimin e figurës (imazhit)

Pastaj, hapet dritarja vijuese, përmes së cilës bashkëngjitet figura:



Figura 8: Zgjedhja e figurës për postim

Përmes opsionit **Choose Files** zgjedhet figura e cila duhet të bashkëngjitet në kompjuter. Figura e selektuar publikohet me klikim në butonin **Add Selected**, që gjendet në këndin e poshtëm majtas nga dritarja:

Ushtrim praktit Post Detyra 1:	📰 🗞
Compose HTML 🔊 🖓 $\mathcal{F}$ T T Vormal V B I U ASC 🗛 $\mathscr{I}$ Link 🖬 🎬 🕲 📑	■ - 注 III 🖆 🎜 💝 - Post settings
Ne programin për llogaritje tabela të krijohet tabela "Radhitje e orarit". Tabela të me fontin Arial, madhësi 12, të zgjidhet ngjyra e kufizimeve (vijave) sipas dëshirë të tabelës të jenë të ngjyrosun që përputhet me ngjyrat e qehive tjera të tabelës. Sinti - Mestum - Lange - Scharge - Originat site   Left - Center - Bje	eete e formatuar s. Qeliate nitullit S. Qeliate nitullit S. Qeliate nitullit S. Published on Big 19, 7.37 AM Paradic Daylight Time or Permaink ♀ Location ★ Options M   Addiantion   Proceeding   Barrow 3]
Eigura 9: Drogullimi (rodaktim	i) i figuräs pär publikim

Shënim!

Përveç bashkëngjitjes së një figure nga kompjuteri, gjatë krijimit të postimit (publikimit) me figura mund të përdoren edhe figurat të cilat tashmë janë përdorur nga blogu, figura nga arkiva e Google, si dhe figura nga telefoni.

Nga dritarja, kur selektoni figurën (imazhin) e shtuar, shfaqen opsionet për editimin e tij, të tilla si: editimi i dimensioneve të figurës, klasifikimi i figurës, shtimi i etiketës në figurë, cilësi shtesë dhe fshirja e tij.

Procedura për të shtuar një video në blog është e njëjtë me shtimin e figurës. Për të shtuar një video në postim, duhet që video të shkarkohet dhe të ruhet në kompjuter. Duke klikuar në opsionin **Choose video to upload**, videoja fillon të bashkëngjitet në postim. Kur ngarkimi të ketë përfunduar, klikoni te **Publish** (publikimi).



### Mundohuni!

Çfarë janë programet? Jepni përkufizime për programimin! Si përpunohet një program? Cilat gjuhë programuese i dini?

### 5.2.4 Shtimi i linqeve në blog



# Të rikujtojmë!

Çfarë janë linqet? Si mund të dalloni nëse një tekst është link? Theksoni disa karaktersitika të linqeve!

Me postimin (**Post**) në blog, përdoruesi mund të shtojë edhe linqe, përkatësisht të lidhet me faqe të tjera të vendosura në Internet. Për këtë qëllim, selektohet teksti në postim i cili duhet të jetë link dhe klikohet në mjetin nga dritarja për shtimin e linqeve:

Ushtrim praktit				Po
Compose HTML	$\mathcal{F}$ $\mathcal{F}$ $\mathbf{T}$ $\mathbf{F}$ Normal $\mathbf{F}$ $\mathbf{B}$	Z <u>U</u> ABC <u>A</u> ▼ ∅ ▼	Link 🖬 🎬 😳 🗟 📑 🕶 🗄 🖽 🏒	130
	Në programin për llogaritje tabela t me fontin Arial, madhësi 12, të zgji të tabelës të jenë të ngjyrosura që p	e Edit Link	ınd të mermi më shumë informata!	×
		Link to:	To what URL should this link go?	
		Web address	www.computersvii.blogspot.com	
		Email address	Test this link	
	2		Not sure what to put in the box? First, find the page of web that you want to link to. (A search engine might be Then, copy the web address from the box in your brows address bar, and paste it into the box above.	on the usefuser's
	Këtu mund të merrni më shumë info	Open this link in a r     Add 'rel=nofollow' a	new window ttribute (Learn more)	
		OK Cano	el	

Figura 10: Shtimi i linqeve për postim (publikim) në blog

Në dritaren për editim të linkut (**Edit Link**), në fushën **Text to Display**, paraqitet teksti i cili është selektuar në postim për të cilin përdoruesi ka vërtetuar se dëshiron të jetë link. Pastaj, selektohet butoni **Web Address** dhe në fushë shkruhet adresa e faqes në të cilën përdoruesi dëshiron të lidhet. Nëse duhet të vendoset lidhja me e-mail adresën, atëherë shtypet butoni **Email Adress** dhe në fund klikohet në butonin OK. Për të publikuar shpalljen klikohet në **Publish**.

#### 5.2.5 Ndryshimi i temës/pamjes së blogut

Gjatë editimit të përmbajtjes së blogut, përdoruesi, megjithëse ka zgjedhur një temë për blogun, mund ta ndryshoj atë. Procedura për ndryshimin e temës/ pamjes së blogut është përmes tabelës së kontrollit (**Dashboard**) dhe opsionit **Theme**:



Figura 11 :Zgjedhja e temës/pamjes së blogut

Pasi të zgjedhet tema/pamja, paraqitet se si do të dukej blogu dhe nëse pamja është e pranueshme, klikojmë në butonin **Add** për të shtuar temën.

Në fund, në tabelën e kontrollit klikohet në linkun **View Blog** që të mundet të bëhet rishikimi i postimeve dhe pamjes së blogut.



Blog-shërbimi më i zakonshëm i përdorur është Blogger dhe i përket Google. Përdoruesi duhet të ketë një llogari Gmail, që përfshinë emrin e përdoruesit dhe fjalëkalim për tu paraqitur. Shpalljet ose postimet janë përmbajtja që publikohet në blog. Nëpërmjet tabelës së kontrollit, autori i blogut mund të krijoj cilësime, të shtojë postime, të modifikoj postime, të ndryshoj tema/ pamjen e blogut, të rishikoj pjesëmarrjen e blogut, etj.



- 1. Theksoni disa shërbime te blogu!
- 2. Cili është shërbimi më i popullarizuar i blogut?
- 3. Pse përdoret tabela e kontrollit (dash table) gjatë krijimit të blogut?
- 4. Çfarë janë postimet?
- 5. Cila është procedura për krijimin të postimit?
- 6. Cila mund të jetë përmbajtja e një blogu?



### 5.3 Komentimi i përmbajtjeve të blogjeve

Siç u tha në fillim të temës, bloget na lejojnë të jemi pjesë e bashkësisë me temë specifike. Në fakt, mund të marrim pjesë në një diskutim për çdo temë nëse tregojmë interes në zgjerimin e njohurive, ndarjen e përvojave, gjetjen e një zgjidhjeje për problemin, demonstrimin e praktikës së mirë etj. Për arsyet e deklaruara, mund të komentojmë mbi postimet, përkatësisht postimet në blog.

Në mënyrë që të jemi në gjendje të komentojmë mbi përmbajtjen e blogut, së pari duhet të thërrasim përmbajtjen e blogut për shqyrtim duke futur adresën e blogut në shiritin e adresave të ueb-shfletuesit. Pastaj, e shqyrtojmë përmbajtjen e tij. Për të shtuar komentin në postim, klikojmë në **Comment**, pastaj shfaqet forma për të shtuar komentin, siç tregohet në figurë:

A duhet	detyra të re	alizohet në Excel	2016 ?	
				,
Co	mment as: (	Google Accour 🔻		
00		Boogle Accourt		

Figura 1: Shtimi i komentit në postimin në blog

Në fushën e zbrazët, shkruani tekstin e komentit dhe zgjedhim profili përmes të cilit do të komentohet. Pastaj ekzistojnë dy mundësi: **Publish** dhe **Preview**. Nëse zgjedhet **Publish**, komenti do të botohet nën përmbajtjen e postimit dhe me **Preview** shqyrtohet rishikimi i komentit.

Administratori ose krijuesi i blogut, që të mund të përgjigjet në komentet e postuara ose të përfshihet në diskutimin e filluar, së pari bën një përmbledhje duke vepruar përmes tabelës së kontrollit (**Dashboard**) dhe opsionit **Comment**.



Figura 2: Rishikimi i komenteve për përmbajtjet në blog

Meklikim në koment mundësohet opsioni **Reply** që t'i mundëson administratorit të blogut t'i jep përgjigje komentit. Në fushën e zbrazët shkruhet përgjigjeja dhe klikohet në **Publish** që të jetë i publikuar.



Kur vendosni blog, edhe përdoruesi edhe administratori shtojnë komente. Përmes komenteve, përdoruesit fillojnë bisedën, ndajnë përvoja, pyesin për gjëra të panjohura, u përgjigjen pyetjeve. Komentet ju lejojnë të jeni pjesë e bashkësisë në tema të ndryshme.



## TË PËRSËRISIM! TË USHTROJMË!



### Detyra 1

Plotësoni diagrami vijues në mënyrë që rregullat e lejuara për përdorimin etik të shërbimeve të Internetit të shkruhen me të gjelbër dhe ato të palejuara me të kuqe.



Detyra 2

Numëroni këshillat për vendosjen e gjurmëve (shtypjes) digjitale pozitive.



### Detyra 3

Sqaroni se çfarë paraqesin ueb-ditarët!



Sqaroni se çfarë paraqesin ueb-ditarët!.

Njerëzit që editojnë ueb-ditarët quhen \_\_\_\_\_\_ ndërsa procedura për editimin (redaktimin) e blogjeve quhet \_\_\_\_\_.

154

Detyra 5

Në ueb-ditarët lajmet/përmbajtjet publikohen sipas:

- a) sipas numrit të figurave të publikuara në blog;
- b) pa ndonjë radhitje të veçantë;
- c) sipas radhitjes kronologjike, përkatësisht sipas datës së publikimit.



### Detyra 6

Para se të fillohet me procedurën për përpunim të blogut:

- a) është e nevojshme që përdoruesi të ketë e-mail adresë postë elektronike;
- b) përdoruesi mund të ketë e-mail adresë, por kjo nuk është e nevojshme;
- c) nuk ka nevojë për e-mail adresë.



### Detyra 7

Emri i përdoruesit i pronarit (autorit të blogut) përdoret si:

- a) emër për nënshkrim;
- b) emër për paraqitje në ueb-ditarin;
- c) tema për diskutim.



### Detyra 8

#### Postimet në blog quhen:

- a) statuse;
- b) postime;
- c) përshkrime.



### Detyra 9

Vizitori i blogut, mund të krijon porosi për një ose më shumë postime në blog përmes opsioni për shtimin:

- a) linkut;
- b) blog-arkivës;
- c) komentet.



### Detyra 10

Çfarë duhet bërë që të publikohet një postim i ri i blogut?

#### Sqaroni kuptimin e çdo elementi!



### Detyra 12

Si të sillemi kur shkruani ose vizitoni blogje të caktuara në Internet?

# Detyrë projektuese

Krijoni ueb-ditarin/blog në ndonjë nga shërbimet pa pagesë të blogjeve. Zgjedhim një temë që korrespondon me ndonjë nga aktivitetet e projektit në shkollë ose që korrespondon me tema nga programi mësimor, të tilla si: eko-integrimi, integrimi ndëretnik dhe ndërkulturalizmi në shkollë, sport dhe aktivitete sportive, art dhe ngjarje, lojë, programim, etj. Zgjedhim me kujdes një temë/pamje të blogut që përputhet me përmbajtjen e blogut. Blogu duhet të përmbajë lloje të ndryshme të përmbajtjes: tekst, linqe, video, imazhe, etj. Këmbeni adresat e blogut me miqtë tuaj në mënyrë që të mund të lini komente mbi postimet, pastaj t'i përgjigjeni pyetjeve si në vijim:

- 1. Cila është adresa e blogut tuaj?
- 2. Cili është titulli i blogut tuaj?
- 3. Për çfarë është i destinuar blogu?
- 4. Në cilën datë është shkruar regjistrimi i fundit në blog?
- 5. Në cilën datë është shënuar regjistrimi i parë në blog?
- 6. Përveç regjistrimeve çfarë informacionesh tjera ka blogu?

### FJALORI I TERMAVE KRYESOR (terminologjia profesionale)

Qeliza aktive	Qeliza e cila për momentin është e klikuar dhe është e gatshme për vendosjen e të dhënave quhet qelizë aktive.	Tema 1
Analiza	Analiza dhe zbërthimi i detyrës problemore në elemente të veçanta	Tema 3
Numrat binarë	Numrat binar janë 0 dhe 1. Këto e përbëjnë gjuhën binare ku të dhënat, informacionet dhe komandat përkthehen në vargun e 0 dhe 1.	
Blog	Blogu ose ueb-ditari paraqet ditarin onlajn ku përmbajtja e radhitjes kronologjike, përkatësisht lajmet më të reja ose përmbajtjet paraqiten në fillim, dhe të tjerat shkojnë më poshtë.	Tema 2
Grafiku	Grafiku paraqet paraqitje grafike të të dhënave.	Tema 5
Paraqitja grafike	Paraqitja me figura (imazh).	Tema 1
Debager	Debageri i gjenë dhe i mënjanon gabimet.	Tema 4
Gjurmë (shtypje) digjitale	Gjurmë digjitale - janë të gjithë informacionet në internet (pozitive dhe negative) të cilat janë vendosur rastësisht ose me qëllim	Tema 5
Editori	Rregullues (editues) i kodit burimor.	Tema 4
Eliminimi	Eliminimi është proces i hudhjes së fakteve që nuk janë të rëndësishme dhe të cilat nuk na çojnë deri në zgjidhjen përfundimtare të situatës problemore.	Tema 3
Regjistrimi (shënimi)	Regjistrimet janë shpalljet në blog.	Tema 5
Programi burimor	Programi burimor është lista e komandave tekstuale të cilat duhet të përkthehen programin ekzekutiv kompjuterik.	Tema 4
Programi ekzekutiv	Programi ekzekutiv është datotekë e cila mundet të ekzekutohet si program në kompjuter.	Tema 4
Inicializimi	Ndarja e vlerave të ndryshores gjatë definimit të saj quhet inicializim.	Tema 4

Programe interaktive	Programet interaktive janë programet në të cilat ekziston ndonjë lloj komunikimi (interaksioni) në mes të subjekteve.	Tema 3
Deklarimet (urdhërat)	Deklarimet paraqesin urdhëra të cilët i tregojnë kompjuterit se çfarë të bëj.	Tema 3
Kodimi	Përkthimi i të dhënave, informacioneve dhe programeve në gjuhë të kuptueshme për kompjuterin dhe anasjelltas quhet kodim.	Tema 2
Kolona	Kolonat janë pjesë vertikale të tabelës.	Tema 1
Komenti	Komentet janë mjete më të thjeshta për interaksion në blog, përkatësisht për vendosjen e komunikimit në bashkësi.	Tema 5
Kompajleri	Kompajleri është përkthyesi i fajllit burimor në ekzekutiv.	Tema 4
Konstantet	Konstanta është madhësi e qëndrueshme dhe e pandryshueshme.	Tema 3
Koordinatat	Elementet gjeometrike të cilat përdoren për vërtetim të saktë të pozitës së ndonjë pike.	Tema 3
Të menduarit logjik	Të menduarit logjik është mënyra për ardhje deri te zgjidhja përmes respektimit të vërtetësisë së verifikimeve, përkatësisht deklarimeve.	Tema 3
Ngjarja	Ngjarja në gjuhën programuese paraqet një element të ri të ndërtuar, i cili mundëson interaktivitet me programin përmes aktiviteteve me mausin (miun) ose me shtypjen e butonit përkatës të tastierës.	Tema 3
Qelizat jo të njëpasnjëshme	Qelizat jo të njëpasnjëshme janë qeliza jo fqinje.	Tema 1
Qelizat e njëpasnjëshme	Janë qeliza të cilat gjenden njëra pas tjetërs ose njëra përkrah tjetrës.	Tema 1
Programimi	Shkruarja e programeve me ndihmën e gjuhëve programuese quhet programim.	Tema 2

Ndryshoret	Ndryshoret ose variablat që ndryshojnë në varësi të të dhënës hyrëse.	Tema 3
Fletorja e punës	Fletoret e punës janë dokumente të cilat krijohen në programin për llogaritje tabelare. Për shembull: MS	Tema 1
	Office Excel, Open Office.org, Apple I work Numbers.	
Lista e punës	Çdo fletore e punës në programin për Ilogaritje tabelare përbëhet nga listat (fletat) e punës.	Tema 1
Rreshtat	Rreshtat janë pjesë horizontale të tabelës.	Tema 1
Softueri i lirë	Softueri i lirë është softueri që mund të përdoret, të përshtatet dhe të ndryshon pa kufizime, që mund të kopjohet dhe të ridistribuohet në formën e ndryshueshme ose të pandryshuesme ose me kufizime minimale.	Tema 2
Shprehje krahasuese	Shprehja krahasuese është urdhër ose deklarim i cili mundëson karahasimin në mes vlerave	Tema 4
Strukturat për të dhënat	Struktura për të dhënat janë grupe të të dhënave të definuar me një emër.	Tema 2
Tabela	Tabela është mënyra e organizimit të të dhënave në kolona dhe rreshta.	Tema 1
Qeliza/celula	Prerja (kryqëzimi) e kolonës dhe rreshtit quhet qelizë ose celulë.	Tema 1
Formulat	Formulat janë shprehje aritmetikore dhe logjike, të cilat i krijon përdoruesi me qëllim të realizoj llogaritje.	Tema 1
Funksionet	Funksionet janë formula të gatshme të cilat përmbajnë emër dhe argument për llogaritje	Tema 1
Cikli	Cikli paraqet përsëritjen e vargut të urdhërave deri në plotësimin e kushtit.	Tema 4

### LITERATURA E PËRDORUR

Davis, S.(2014). Beginning Programming with C++ for Dummies. Wiley Publishing

Дејтел П, Дејтел Х. (2010). С++ Како се програмира-седмо издание. Арс Ламина

Freun S, Starks J, Schmieder E. (2016). Microsoft Office 365 Excel 2016 comprehensive. University Indianapolis

Gardner S. (2005). Buzz Marketing with Blogs for Dummies. Wiley Publishing

Marji M. (2014). Learn to program with Scratch. William Pollock

www.bebras.org

https://education.microsoft.com

www.bbc.co.uk

www.code.org