

Do sada ste upoznali program za pisanje i program za crtanje. Međutim, možete napisati i vlastiti program. Naučit ćete kako računalom rješavati matematičke zadatke, a u idućim razredima naučit ćete programirati jednostavnije igrice i crtati računalom.



algoritam, algoritam slijeda, algoritam grananja, naredba, programiranje, program, dijagram toka, objekt, operacija, svojstvo

1. Moj prvi algoritam

Programima *Bojanje* i *Word* crtali ste crteže i oblikovali tekst. Ti programi napisani su u nekome programskom jeziku. I vi možete naučiti pisati programe koje će računalo izvršavati. Da bismo napisali program koji rješava neki problem (npr. zbraja dva broja ili ih uspoređuje), moramo znati opisati taj problem na način koji računalo razumije.

Algoritam je postupak kojim se opisuje točan redoslijed kojim obavljamo neki posao.

Evo primjera algoritma za slaganje školske torbe:

1. Uzmi raspored sati.
2. Stavi knjige i bilježnice za svaki predmet u torbu.
3. Stavi potreban pribor u torbu.
4. Provjeri još jednom raspored sati i uvjeri se je li sve u torbi.
5. Stavi papuče i torbu pored vrata.

Vježba 1.

Otvorite kviz  [56Bkviz1.htm](#). Riješite nekoliko primjera iz svakodnevnoga života u kojima korake algoritma morate složiti u pravilan redoslijed.

▶ Algoritam je postupak kojim se opisuje točan redoslijed kojim obavljamo neki posao.

U svakodnevnom životu često se koristimo algoritmima kako bismo obavili različite poslove, a da toga nismo ni svjesni. Jedan takav primjer pisanje je zadaće.

Algoritam za pisanje zadaće iz matematike

početak

potraži bilježnicu i udžbenik

otvori bilježnicu i udžbenik

pripremi našiljenu olovku i gumicu

pronađi zadatke koji su za zadaću

pročitaj ih

ako nešto ne razumiješ, onda

prouči iz bilježnice što ste učili u školi

pronađi u udžbeniku slične riješene zadatke

riješ zadane zadatke

provjeri na kraju udžbenika jesi li točno riješio zadatke

ako imaš pogrešaka, **onda**

ispravi pogreške

ako ne znaš ispraviti, **onda** sutra pitaj učiteljicu

inače

zadaća je gotova

kraj

Algoritam sastavljamo od točno određenih riječi kojima poslije pišemo program. Napisani program sadrži **naredbe** koje su razumljive računalu.

▶ Naredba je uputa računalu da obavi jednu operaciju.

Naprimjer, evo nekoliko naredbi:

upiši x

ispiši x

y=x+5

ako je $y > x$, **onda ispiši** x, **inače ispiši** y

Algoritmom detaljno opisujemo problem koji računalo treba riješiti. Da bismo pisali svrhovite programe, moramo točno predvidjeti što želimo da računalo napravi.

▶ **Program** je konačan niz naredbi razumljivih računalu koje rješavaju neki problem. Postupak pisanja programa zovemo **programiranje**.

Program pišemo u nekome **programskom jeziku**. Kao što mi međusobno razgovaramo hrvatskim jezikom, računala međusobno "razgovaraju" programskim jezicima.



Algoritam je stara riječ koja potječe iz arapskoga jezika, a znači *postupak, pravilo, uputa*.

Algoritam - postupak, pravilo, uputa

program, programiranje

Računalo podatke pamti na određenome mjestu u svojem spremniku. To mjesto zovemo memorijsko mjesto ili memorijska lokacija (od engl. *memory location*).

► **Programski jezik** skup je naredbi i pravila za njihovo pisanje razumljivih računala.

Programski jezik ima mnogo manje naredbi i mnogo jednostavnija pravila za njihovu upotrebu nego bilo koji govorni jezik. Neki programski jezici namijenjeni su upravo djeci, takvi su npr. Logo ili Basic. U ovome poglavlju učiti ćete programski jezik **Small Basic**. Sada ćemo još malo učiti pisati algoritme, a kasnije ćemo za te algoritme pisati programe u Small Basicu.

Napišimo algoritam za zbrajanje dvaju brojeva.

upiši a
upiši b
zbroj = a+b
ispiši zbroj

Opišimo algoritam riječima:

1. zadajemo jedan broj koji će računalo zapamtiti u memorijskome mjestu nazvanome **a**
2. zadajemo drugi broj, a računalo će ga zapamtiti u memorijskome mjestu nazvanome **b**
3. program računa zbroj i sprema ga u memorijsko mjesto nazvano **zbroj**

4. na zaslonu će se ispisati broj koji je spremljen u mjestu **zbroj**.

Što će algoritam izračunati ako je prvi broj 10, a drugi broj 5?

Algoritam u memorijsko mjesto **a** stavlja 10, a u memorijsko mjesto **b** stavlja 5. Onda se u mjesto **zbroj** stavlja 15, a na zaslonu se ispisuje 15.

Što će algoritam izračunati ako je prvi broj 365, a drugi broj 956?

Algoritam u memorijsko mjesto **a** stavlja 365, a u memorijsko mjesto **b** stavlja 956. Onda se u memorijsko mjesto **zbroj** stavlja 1 321, a na zaslonu se ispiše 1 321.

Vidimo da taj algoritam može zbrojiti bilo koja dva broja.

Evo još jednoga primjera. Otac je djeci obećao 5 kn džeparca za odlazak u trgovinu i 10 kn za usisavanje stana. Djecu zanima koliko će mjesečno dobiti džeparca.

- a) Koliko su novca djeca zaradila ako su 10 puta bila u trgovini i 4 puta usisala stan?
- b) Jesu li djeca više zaradila ako su 6 puta bila u trgovini i 2 puta usisala stan ili ako su 2 puta bila u trgovini i 4 puta usisala stan?
- c) Ako djeca žele zaraditi 100 kn, a ne žele ići u trgovinu, koliko puta moraju usisati stan? Napišite algoritam koji to računa.

2. Od algoritma do dijagrama toka

Algoritam se može prikazati i crtežom, odnosno grafički. Grafički prikazan algoritam nazivamo **dijagram toka**. Pri tome geometrijski likovi predstavljaju pojedinu vrstu naredbe.

► Grafički prikaz algoritma jest **dijagram toka**.

Nacrtajmo dijagram toka za zbrajanje dvaju brojeva:

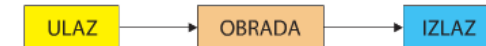
Geometrijski oblici u dijagramu toka ovise o vrsti naredbe koju prikazuju. Tako prikazan algoritam vrlo je pregledan i potpuno određen. Posebno je pogodan za analizu programa, traženje sličnih rješenja ili potrebne izmjene.

Ovalni oblik služi za oznaku početka ili kraja programa.

Četverokut u obliku lijevka koristi se za ulaz ili za izlaz podataka.

U pravokutnik pišemo naredbe kojima ćemo npr. nešto izračunati.

Postupak programiranja sastoji se od triju osnovnih koraka:



Za problem koji želimo riješiti računalom najvažnije je pronaći dobar algoritam. Pod pojmom "dobar" mislimo na brz i učinkovit algoritam.

Vježba 2.

Napišite algoritam i nacrtajte dijagram toka za računanje opsega i površine školskoga igrališta.

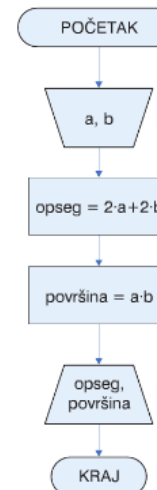
Rješenje:

Školsko je igralište pravokutnik. Ako zadamo duljinu i širinu igrališta, lako ćemo izračunati opseg i površinu igrališta.

algoritam:

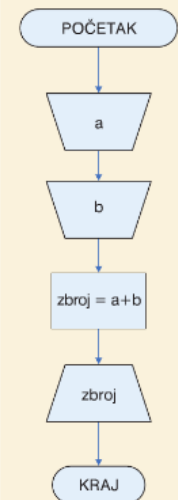
upiši a,b
opseg = 2 · a + 2 · b
površina = a · b
ispiši opseg, površina

dijagram toka:



Dijagram toka – grafički prikaz algoritma

dijagram toka:



Četverokut u obliku lijevka zove se trapez. To ćete naučiti na satu matematike.

Slijedni algoritam ili algoritam slijeda

3. Kako donosimo odluke

Do sada smo naučili pisati algoritme u kojima naredbe slijede jedna iza druge i tako se redom izvršavaju. To su **algoritmi slijeda** u kojima se naredbe izvršavaju redom kako su napisane.

▶ Ako se u algoritmu naredbe izvršavaju jedna iza druge, kažemo da je to **slijedni algoritam** ili **algoritam slijeda**.

Međutim, računalo često mora odlučiti koju će naredbu izvršiti, ovisno o problemu koji rješava. Ponovimo neke naredbe iz *Algoritma za pisanje zadaće*: **ako** imaš pogreška **onda**

ispravi pogreške
inače
zadaca je gotova

Iz gornjega primjera razvidno je da se naredbe u algoritmu ne moraju uvijek sve izvršiti. Primjerice, ako imamo pogreška u zadaci, onda ih moramo ispraviti. Ali ako nema pogreška, zadaću smo završili. Pretpostavimo da je mama rekla Ivici da si na putu do škole od dječarca može kupiti sok. Ivica može kupiti sok samo ako ima 5 kn. Što ako Ivica nema 5 kn? Kako bi tada glasila naredba?

Npr. može glasiti ovako:

ako imaš 5 kn, **onda** kupi sok, **inače** nastavi u školu

To znači da je *imaš 5 kn* **uvjet**. Ako je uvjet istinit (tj. ti imaš 5 kn), tada se izvršava naredba iza **onda**, što znači da Ivica može kupiti sok. Ako uvjet nije istinit (tj. Ivica nema 5 kn), izvršava se naredba iza **inače** (znači da mora nastaviti u školu). Jasno je da se sok ne može kupiti s manje od 5 kn.

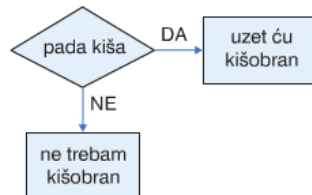
Što ako Ivica ima 6 ili 7 kn? Može li onda kupiti sok? Kako bi glasila takva naredba? U tome slučaju napisali bismo:

ako imaš barem 5 kn, **onda** kupi sok, **inače** nastavi u školu

Kako ćemo u dijagramu toka prikazati naredbu odluke poput sljedeće:

ako pada kiša, **onda** uzet ću kišobran, **inače** ne trebam kišobran?

U dijagramu toka algoritam odluke prikazujemo **rombom**, pa ćemo naredbu o kiši nacrtati ovako:



Uvjet u algoritmu

Symbolima dijagrama toka nacrtajte sljedeće naredbe:

ako nemam zadaće, **onda** se mogu igrati, **inače** moram napisati zadaću
ako prijatelj treba pomoć, **onda** pomoz prijatelju, **inače** se možeš igrati

Dakle, u romb upisujemo uvjet. Zatim se algoritam nastavlja u jednome od dvaju smjerova. Ako je uvjet istinit, izvršava se naredba koja slijedi iza strelice DA, a ako uvjet nije istinit, izvršava se naredba koja slijedi iza strelice NE.

Takav algoritam zovemo **algoritam grananja**, jer se iza naredbe odluke algoritam nastavlja ili grana u jednome od dvaju mogućih smjerova.

Algoritam grananja

▶ Ako se u algoritmu nalazi naredba odluke, oblika **ako ... onda ... inače**, takav algoritam zove se **algoritam grananja** – iza naredbe odluke algoritam se nastavlja u jednome od dvaju opisanih smjerova.

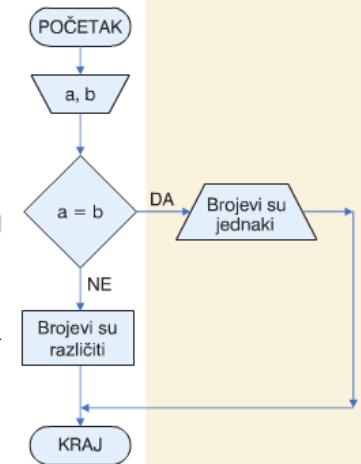
Vježba 3.

Napišite algoritam koji će provjeriti jesu li dva broja jednaka ili nisu.

algoritam

dijagram toka:

upiši a,b
ako je a=b **onda**
 ispiši Brojevi su jednaki.
inače
 ispiši Brojevi su različiti.



Kako biste napisali algoritam koji će nam reći koji je od dvaju zadanih brojeva veći, a koji je manji?

Vježba 4.

Napišite algoritam koji uspoređuje dva broja **a** i **b**. Iz matematike znamo da postoje samo tri mogućnosti:

1. jedan broj može biti manji od drugoga ($a < b$)
2. jedan broj može biti veći od drugoga ($a > b$)
3. brojevi su jednaki ($a = b$).

Imamo tri mogućnosti, pa nam jedna naredba odluke neće biti dovoljna. Pogledajte algoritam i dijagram toka.

algoritam

upiši a,b
ako je a>b **onda**
 ispiši a > b
inače ako je a<b **onda**
 ispiši a < b
inače
 ispiši a = b.



Programski jezik Small Basic jedna je od inačica programskoga jezika BASIC. Riječ BASIC složena je kratica od **B**eginners **A**ll-purpose **S**ymbolic **I**nstruction **C**ode, (programski jezik različitih namjena za početnike)

dijagram toka:

1. Upišemo dva broja **a** i **b**.
 2. Ako je $a > b$, onda ispisujemo $a > b$ i idemo na kraj dijagrama toka.
 3. Ako **a** nije veći od **b**, moramo provjeriti je li $a < b$. Tada ako je $a < b$, ispisujemo poruku $a < b$.
 4. Na kraju, ako nije $a < b$, a već znamo da nije $a > b$ ostaje samo mogućnost da je $a = b$, pa ispisujemo $a = b$.
- Dakle, za uspoređivanje dva broja potrebne su dvije naredbe odluke.

