

Računske operacije

1 Seštevanje, odštevanje, množenje

Računalniki lahko s spremenljivkami počnejo veliko stvari. Najpreprostejše so operacije na številih, kot so seštevanje, odštevanje in množenje. Račune zapisujemo tako kot v šoli, z *operatorji*.

- + za seštevanje
- - za odštevanje
- * za množenje

Rezultat lahko izračunamo kar znotraj funkcije `printf`:

Primer

```
#include<stdio.h>

int main(){
    int a = 5, b = 7;
    printf("%d\n", a+b);
    return 0;
}
```

Primer vhoda in izhoda

12

Rezultat lahko tudi shranimo v novo spremenljivko:

Primer

```
#include<stdio.h>

int main(){
    int a, b, vsota, razlika, produkt;
    scanf("%d%d", &a, &b);
    vsota = a+b;
    razlika = a-b;
    produkt = a*b;
    printf("%d\n%d\n%d\n", vsota, razlika, produkt);
    return 0;
}
```

Primer vhoda in izhoda

3 7

12

-4

21

Pogoste napake

Spremenljivko *inicializiramo* tako, da notri nekaj shranimo, bodisi kot `a=5`, bodisi s tem, da vanjo nekaj napišemo s funkcijo `scanf`. Če je ne inicializiramo, pozneje pa jo poskusimo uporabiti za izpisovanje, računanje ali kaj drugega, lahko dobimo zelo čudne rezultate, naš program se lahko celo sesuje.

V zgornjem programu tudi vidimo, da lahko z enim klicem funkcije `scanf` preberemo več spremenljivk.

Negativna števila

Na meteorološki postaji Kredarica so leta 2014 izmerili povprečno januarsko temperaturo približno -5 °C, povprečno avgustovsko pa približno 6 °C. Med tema meritvama je 11 °C razlike.

Pozimi lahko izmerimo temperature manjše od 0. Takšnim številom, kot je -5 , rečemo *negativna števila*. Lahko jih uporabimo tudi drugje, ne samo pri merjenju temperature.

S pozitivnimi števili lahko štejemo od 0 do neskončno (1, 2, 3, ...), z negativnimi pa do - neskončno. (-1 , -2 , -3 , ...). Tako kot pozitivna števila jih lahko seštevamo in odštevamo:

Primer

$$\begin{aligned}5 - 11 &= -6 \\ -6 + 11 &= 5 \\ 5 - (-6) &= 11 \\ -2 - 1 &= -3 \\ -2 - (-1) &= -1 \quad -2 + (-1) = -3\end{aligned}$$

Pravila:

$$\begin{aligned}-(-6) &= +6 \\ +(-1) &= -1 \\ -(-(-(-1))) &= -(-(+1)) = -(-1) = 1\end{aligned}$$

Lahko jih tudi množimo:

Primer

$$\begin{aligned}2 * (-5) &= -10 \\ (-5) * (-5) &= 25\end{aligned}$$

Pravila: Če množimo dve pozitivni števili, je produkt pozitiven.
Če množimo eno pozitivno in eno negativno število, je produkt negativen.
Če množimo dve negativni števili, je produkt spet negativen.
O negativnih številih lahko razmišljamo kot: $-3 = (-1) * 3$

Računalniki z negativnimi števili računajo enako kot s pozitivnimi:

Primer

```
#include<stdio.h>

int main(){
    int a, b;
    scanf("%d%d", &a, &b);
    printf("Vsota: %d\nRazlika: %d\nProdukt: %d\n", a + b, a - b, a * b);
    return 0;
}
```

Primer vhoda in izhoda

3 7

Vsota: 10
Razlika: -4
Produkt: 21

2 Deljenje

Števila lahko tudi delimo. Za deljenje uporabljamo znak /. Za razumevanje poglavja si bomo pomagali s formulo $a = k * b + o$, ki ponazarja deljenje z ostankom.

Primer

```
#include<stdio.h>

int main(){
    int a, b;
    scanf("%d%d", &a, &b);
    printf("%d\n", a/b);
    return 0;
}
```

Primer vhoda in izhoda

16 8

2

Pogoste napake

Deljenje v jezikih C in C++ je celoštevilsko. Deljenje z ostankom lahko zapišemo po formuli:
 $a = k * b + o$.

16/8: $16 = 2 * 8 + 0$

Ostane je 0, ker je 16 deljivo z 8.

16/5: $15 = 3 * 5 + 1$

1 je ostanek.

Celoštevilsko deljenje pomeni, da nam program vrne samo k . Primer vhoda in izhoda za zgornji program, kjer števili nista deljivi:

Primer vhoda in izhoda

16 3

5

Tudi, če bi bil rezultat deljenja 7.9, tega program ne zaokroži na 8, temveč nam vrne 7. Če manjše število delimo z večjim, zato vedno dobimo 0.

Operator / nam torej iz pri deljenju a/b vrne k . Lahko pa dobimo tudi ostanek o . Do njega pridemo z operatorjem % (*modulo*).

Primer

```
#include<stdio.h>

int main(){
    int a, b;
    scanf("%d%d", &a, &b);
    printf("%d = %d*%d + %d\n", a, a/b, b, a%b);
    printf("Količnik: %d\nOstanek: %d\n", a/b, a%b);
    return 0;
}
```

Primer vhoda in izhoda

25 7

25 = 3*7 + 4

Količnik: 3

Ostanek: 4

Pogoste napake

Deljenje z 0 v matematiki ni definirano in prav tako ne v programiranju. Če neko število delimo z 0, se nam bo program sesul. Prav tako, če poskušamo izračunati ostanek pri deljenju z 0. Ta napaka se pogosto zgodi v programih, kjer delimo z več števili, zato moramo biti na to pozorni.