

Računske vježbe 1

Programiranje I

1. Koji od navedenih identifikatora nisu ispravni i zašto?

aaa	_1_
a12	Windows
12a <u>1</u>	vece-manje&drugo <u>1</u>
_12a	double <u>1</u>

Imena, odnosno identifikatori koriste se za imenovanje promjenljivih, funkcija i makroa. Imena se mogu sastojati od slova, cifara i znaka podvlaka (engl. *underscore*) _. Ime ne može početi cifrom, ali može slovom ili podvlakom. Takođe, ime ne može biti jednako ključnoj riječi.

2. Koju vrijednost ima promjenljiva `b` poslije izvršavanja navedenih naredbi?

```
short int a=32767, b;  
b=a+1;
```

U programskom jeziku C, sve promjenljive pripadaju jednom od 10 tipova podataka. Sve tipove podataka karakterišu memorija koju zauzimaju, operacije koje se nad njima mogu izvršavati i domen, odnosno oblast definisanosti. Promjenljive `a` i `b` definisane su kao `short int` pa znamo da su vrijednosti koje se u ove promjenljive mogu upisati u opsegu od -32768 do 32767 . Kako je promjenljiva `a` inicijalizovana na 32767 , što je poslednja vrijednost u zadatom opsegu, a kako se u promjenljivu `b` mogu upisati samo vrijednosti iz istog opsega, nakon inkrementiranja vrijednosti smještene u promjenljivoj `a` i dodjeljivanjem te vrijednosti promjenljivoj `b`, u njoj će se naći -32768 , što je prva vrijednost iz opsega.

3. Koju će vrijednost imati realna promjenljiva `B` poslije datog niza naredbi? Na koji način se u tu promjenljivu može upisati tačan rezultat date operacije?

```
int A;  
float B;  
A=5;  
B=5*A/7;
```

Rezultat operacija koje se izvršavaju nad promjenljivima (operandima) u računarstvu, za razliku od matematike, zavisi od oblasti definisanosti samih promjenljivih. Ukoliko su operandi istog tipa, onda i sam rezultat mora biti tog tipa. Kada su operandi različitog tipa, rezultat će biti onog tipa čija je oblast definisanosti veća. Izraz `5*A/7` uključuje dvije konstante (`5` i `7`, tipa `int`), jednu promjenljivu `A` i dvije operacije. Kako su svi operandi tipa `int`, tako će i međurezultat biti tog tipa, bez obzira što je s lijeve strane izraza promjenljiva tipa `float`. Dakle, vrijednost promjenljive `B` će biti rezultat $5 * 5 / 7$ što je 3 jer se u međurezultat tipa `int` ne može upisati necjelobrojni dio operacije iza zareza. Istine radi, u `B` će biti upisano `3.0`. Tačan rezultat se može upisati na više načina. Jedan od njih je da izmijenimo izraz tako da neku od konstanti zapišemo u formatu vrijednosti s pomičnim zarezom (npr. `5.0` ili `7.0`) ili eksplicitnom konverzijom upotrebom `cast` operatora koji možemo primijeniti kako na promjenljivim, tako i na konstantama:

```
B=5*(float)A/7;
```

4. Koju će vrijednost uzeti promjenljive `s1` i `s2` nakon izvršenja svake od datih naredbi? Koju će vrijednost uzeti promjenljiva `s2` ako se u posljednjem izrazu izostavi spoljašnji par zagrada u drugom sabirku?

```
int s1, s2, q=8;
s1 = 1;
s2 = ++s1 + 5; // s1=2, s2=7
s1 = !s2; // s1=0, s2=7
s2 += (q%3) + ((q<10) || (s1--));
```

`++s1` - prefiksno inkrementiranje: prvo ga povećaj za 1, a zatim ga upotrijebi u izrazu

`!s2` - operator negacije (od izraza različitog od 0 daje 0, i od 0 daje 1)

`q%3` - ostatak pri dijeljenju (isključivo se primjenjuje kod cijelih brojeva)

`s1--` - postfixno dekrementiranje: prvo ga upotrijebi u izrazu, a zatim ga umanj za 1

`||` - logičko ILI (OR), ukoliko je lijevi operand logički tačan, desni se ni ne razmatra

Posmatrajmo sada posljednju liniju koda. Svedimo izraz na dva operanda, `a+b`, gdje je `a=(q%3)` i `b=((q<10) || (s1--))`. Vrijednost ostatka pri dijeljenju 8 sa 3 je 2. Vrijednost `b` može biti 1 ili 0, zavisno od toga da li je izraz tačan. Kako je `q<10` tačno, desni operand operacije logičkog ILI se neće ni razmatrati, tj. `s1` se neće dekrementirati (zadržava staru vrijednost), a vrijednost `s2` biće $7 + 2 + 1$, odnosno 10.

Kada se ukloni spoljašnji par zagrada izraz možemo svesti na `a|b` gdje je `a=(q%3)+(q<10)` i `b=(s1--)`. Kako je `a` logički tačno, vrijednost `s2` biće $7 + 1$, odnosno 8.

5. Odrediti vrijednost promjenljivih poslije izvršenja svake od datih naredbi.

```
int t, a=3, b=7, i, j;
t = a&b;
i = t==3;
j = i==0 ? i : t;
```

`a&b` - logička operacija I (AND) nad bitovima (bit rezultata je 1 samo ako su biti na istom mjestu u `a` i `b` jednaki 1)

`t==3` - provjera jednakosti

`a ? b : c` - ternarni operator: ukoliko je `a` tačno, rezultat je `b`, odnosno `c` ukoliko nije

Pošto znamo da je $3_{10} = 011_2$ i $7_{10} = 111_2$ logička operacija I nad njihovim bitovima je 011_2 odnosno 3_{10} . Kako je vrijednost `t` sada 3, onda je i izraz `t==3` tačan pa je vrijednost `i` sada 1. Logički uslov `i==0` nije tačan, pa će rezultat ternarnog operatora biti `t`.