

Konverzije, operatori, matematičke funkcije

predavač: Nadežda Jakšić

Programiranje programski jezik C

Programi

1. temperatura u stepenima Celzijusa i stepene Farenhajta
 $tf=1.8*tc+32$
2. pretvoriti inče u centimetre
1 inč=2.54cm
3. vreme učitano u satima, minutima i sekundama prikazati u sekundama
4. menjačnica – pretvoriti evre u dinare, uneti dnevni kurs
5. neto i bruto dohodak – uneti cenu rada, broj radnih sati i procenat koji se odbija od bruto dohotka
bruto=cenaRada*brojRadnihSati
neto=umanjiti bruto za procenat

Konverzije

aritmetički izraz

c=a+b

a,b,c su operandi

= ,+ su opertori

- ako su operandi različitih tipova, onda pri izračunavanju dolazi do konverzije operanada u zajednički tip
- biće konvertovani u najprecizniji tip koji je prisutan među operandima i rezultat izraza će biti tog tipa
- ako je x tipa double, a n tipa int, u izrazu x+n pre sabiranja će doći do konverzije promenljive n u tip double i vrednost izraza će biti double

Konverzije

- **short** i **char** (sa pedznakom ili ne) konveruju se u **int**
- ako je **short** isto što i **int**, onda će **unsigned short** biti širi od **int**, pa će operandi biti konvertovani u **unsigned int**
- u svakoj operaciji koja uključuje operande različitih tipova, pre izvršenja operacije vrši se konverzija operanada u precizniji tip
- tipovi su prema preciznosti poređani na sledeći način:

long double

double

float

unsigned long long

long long

unsigned long

long

unsigned int

int

- manje precizni tipovi (char, unsigned char) se ne pojavljuju zato što se oni automatski konvertuju u **int** ili **unsigned int**

Bruto i neto dohodak

```
#include <stdio.h>
#include <conio.h>
int main ()
{
    int  sati, cena, procenat;
    float bruto, neto;

    printf("\n\nUnesite broj radnih sati: ");
    scanf("%d",& sati);
    printf("\nUnesite cenu u dinarima po satu: ");
    scanf("%d",& cena);
    printf("\nUnesite procenat doprinosa koji se odbija: ");
    scanf("%d",& procenat);
    bruto=cena*sati;
    neto=bruto-bruto*procenat/100; //konverzija u precizniji tip
    printf("\nBruto dodatak je %10.2f dinara\n",bruto);
    printf("Neto dodatak je %10.2f dinara\n", neto);
}
```

Konverzije

- konverzija se dešava i kod operacije dodele ako su operandi različitih tipova (operand sa desne strane znaka jednakosti se konvertuje u tip operanda sa leve strane znaka jednakosti)
- pritom može doći do gubitka informacija, ako se konvertuje širi u užu tip (npr. ako je **x** tipa **float**, a **n** tipa **int**, ako napišemo izraz **n=x**, dolazi do odsecanja decimala u promenljivoj **x**)
- eksplicitna konverzija (operator **cast**)
(tip) izraz
pr. double x;
float y;
x=(double)y; //promenljiva **y** je konvertovana u tip **double**

Konverzije

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int a=5;
    printf ("A=%d\n",a);
    int x=5,y=7;
    printf ("zbir je %d\n",x+y);
    float b=6,c=5;
    int k;
    k=b/c;
    printf ("kolicnik je %d\n",k); //k=1 zato što float smeštamo u int
    float m=5.56;
    printf ("m=%f\t m=%d",m,m); //drugi prikaz promenljive m nije dobar
    getch ();
    return 0;
}
```

zato što broj kome su potrebna 4 bajta (**float**) konvertujemo u broj koji je dužine 2 bajta (**int**)

Operatori

aritmetički

+ sabiranje

- oduzimanje, negativan predznak

* množenje

/ deljenje

% moduo broja; celobrojni ostatak kod deljenja (primenjuje se samo za operande tipa **int**)

Aritmetički operatori

```
int a=10, b=4;
float rez;
printf ("a je %d i b je %d\n", a, b);
rez=a+b; //sabiranje
printf ("a+b je %f\n", rez);
rez=a-b; //oduzimanje
printf ("a-b je %f\n", rez);
rez=a*b; //množenje
printf ("a*b je %f\n", rez);
rez=a/b; //mora eksplicitna konverzija (float)a/b
printf ("a/b je %f\n", rez);
rez=a%b; //moduo
printf ("a%%b is %f\n", rez);
```

Operatori

operacije nad bitovima

<< logičko pomeranje ulevo

>> logičko pomeranje udesno

& logičko I za bitove u nizu

~ potpuni komplement

^ isključivo logičko I

| logičko ILI za bitove u nizu

operator CAST

eksplicitna konverzija tipova,
unarni operator
(tip podatka) izraz

```
unsigned char a = 5, b = 9;  
// a = 5(00000101), b = 9(00001001)  
printf ("a = %d, b = %d\n", a, b);  
printf ("a&b = %d\n", a&b);  
// rezultat je 00000001  
printf ("a|b = %d\n", a|b);  
// rezultat je 00001101  
printf ("a^b = %d\n", a^b);  
// rezultat je 00001100  
printf ("~a = %d\n", a = ~a);  
// rezultat je 11111010  
printf ("b<<1 = %d\n", b<<1);  
// rezultat je 00010010  
printf ("b>>1 = %d\n", b>>1);  
// rezultat je 00000100
```

Operatori

elementarni operator dodele

= znak jednakosti

složeni operatori dodele

$a+=3$ je $a=a+3$

$b-=a$ je $b=b-a$

$k*=4+1$ je $k=k*(4+1)$

$x/=2$ je $x=x/2$

operatori umanjivanja, odnosno povećanja vrednosti

-- operator umanjivanja vrednosti **promenljiva se uvek umanjuje ili**

++ operator uvećavanja vrednosti **uvećava za jedan; operatori se**

++k je $k=k+1$ **koriste kao prefiksni (prvo se menja vrednost**

k++ je $k=k+1$ **operanda, a zatim se vrši operacija nad promenlj.)**

j--k prvo $k=k-1$, a zatim $j=k$ **ili postfiksni (prvo se izvršava operacija**

j=k-- prvo $j=k$, a zatim $k=k-1$ **nad promenljivom, a zatim se menja**

vrednost operanda

Operatori

```
#include <stdio.h>
int main()
{
    int x = 10, a;
    a = ++x;
    printf ("pefiksna notacija\n");

    printf ("a= %d",a); //a je 11
    printf ("\nx= %d",x); //x je 11
    return 0;
}
```

```
#include <stdio.h>
int main()
{
    int x = 10, a;
    a = x++;
    printf ("Postfiksna notacija\n");
    printf ("a=%d\n",a); //a je 10
    printf ("x=%d",x); //x je 11
    return 0;
}
```

Operatori

sizeof operator

pomoću ovog operatora može se odrediti veličina memorije koja je potrebna za smeštanje podataka, izražena u bajtovima

unsigned c,d;

c=sizeof (char); **vrednost promenljive c je jedan**

d=sizeof (c); **vrednost promenljive d je 2 (unsigned int zauzima 2 bajta)**

```
#include<stdio.h>
int main()
{
printf ("%d\n",sizeof(char)); //1
printf ("%d\n",sizeof(int)); //2
printf ("%d\n",sizeof(float)); //4
printf ("%d", sizeof(double)); //8
return 0;}
```

```
#include<stdio.h>
int main()
{
int a = 0;
double d = 10.21;
printf ("%d", sizeof(a+d));//8 sabiraju se vrednosti i smeštaju u double
printf ("%d",sizeof(a)+sizeof(d));//12
return 0;}
```

Matematičke funkcije

uključiti zaglavlje `<math.h>`
vrednost svih funkcija je `double`

sin(x) - sinus ugla – vrednost u radijanima

cos(x) – kosinus

tan(x) – tangens ugla

asin(x) – arc sin (x)

acos(x) – arc cos (x)

exp(x) - e^x

pow(x,y) - x^y

sqrt(x) – kvadratni koren

Programi

1. kvadratni koren unetog broja
2. aritmetička sredina tri cela broja
3. vremenski interval koji je dat u minutima izražava se u satima i minutima
4. izdvojiti cifre dvocifrenog broja i ispisti unazad
5. vrednost nekog artikla je x dinara, odrediti najmanju količinu novčanica po 100, 20 i jednog dinara kojima se može platiti taj artikal (npr. za 386 dinara potrebne su 3 novčanice od 100 dinara, 4 novčanice od 20 dinara i 6 novčanica od jednog dinara)

Trougao

6. određivanje dužine stranica trougla ako su zadate koordinate njegovih temena

```
#include <stdio.h>
#include <math.h>
#define PI 3.141592
main()
{
double xa, ya, xb, yb, xc, yc, a, b, c, alfa,
beta, gama;
int izraz =0;
printf ("Unesite koordinate tacke A(x,y):\n
");
scanf ("%lf%lf",&xa,&ya);
printf ("Unesite koordinate tacke B(x,y):\n
");
scanf ("%lf%lf",&xb,&yb);
printf ("Unesite koordinate tacke C(x,y):\n
");
scanf("%lf%lf",&xc,&yc);
```

```
//izračunavanje stranica
a=sqrt(pow(xb-xc,2)+pow(yb-yc,2));
b=sqrt(pow(xa-xc,2)+pow(ya-yc,2));
c=sqrt(pow(xa-xb,2)+pow(ya-yb,2));
printf ("Stranica a je: %.2f\n", a);
printf ("Stranica b je: %.2f\n", b);
printf ("Stranica c je: %.2f\n", c);
return 0;
}
```