

Потсетување за функции од математичката библиотека и кориснички функции

Функции во C++

Потпрограмите се издвоени целини во програма кои решаваат некоја задача. За разлика од некои програмски јазици (на пр. Pascal) кои разликуваат процедури и функции, во C++ се користат само функции. Кај сите програми кои ги разгледавме досега, наредбите кои требаше да се извршат ги пишувавме во една функција – main(). Тоа е основниот дел кој го содржат сите програми, а main() е првата функција која се повикува при стартот на една програма напишана во програмскиот јазик C++.

Функциите претставуваат множества наредби кои се извршуваат при секој нивен повик. Можно е:

- да се напишат сопствени функции па тие да се користат во програма или
- да се користат веќе постоечки функции кои се подготвени за користење и зачувани во библиотеки.

Функцијата треба да изврши една добро дефинирана задача. За да го направи тоа, функцијата може, но не мора да побарува вредности кои ѝ се доставуваат кога таа ќе се повика. Откако функцијата ќе ја заврши задачата, таа во повикувачката функција најчесто враќа некоја вредност.

Што значи враќање на вредноста од страна на функцијата ќе објасниме преку едноставен математички пример.

Променливата c во изразот $c=a+b$ за $a=2$ и $b=3$ ќе добие вредност $c=2+3=5$. Оваа задача може да ја изврши и некоја функција, на пр. `soberi(a,b)` која собира два броја. Името на функцијата е `soberi`, а нејзините параметри се променливите a и b . Ако на параметрите на функцијата `soberi(a,b)` им се доделат вредности $a=2$ и $b=3$, нејзиниот резултат ќе биде 5. Токму овој резултат претставува „повратна вредност“. Значи, кога се вели дека функцијата враќа вредност, тоа значи дека функцијата ја извршила задачата која ѝ е доделена (во овој случај собирање два броја) и го вратила резултатот со кој може да се работи нешто друго или да се прикаже на екранот.

Вредностите коишто ѝ се доделуваат на некоја функција се нарекуваат **аргументи на функцијата**, а вредноста којашто функцијата ја враќа се нарекува **повратна вредност на функцијата**. Повратната вредност во повикувачката функција може да се користи во која било наредба.

Во C++ постојат **функции кои враќаат една вредност и функции кои не враќаат вредност** (на пр. печатење 10 ѕвездички), **но не постои функција која враќа повеќе од една вредност**.

Декларирање функции

Исто како променливите, и функциите треба да се декларираат, т.е. да се најават и еднозначно определат. При декларирањето на функцијата се наведува нејзиното име, тип и листа на параметри. Името на функцијата треба да биде такво за да означува што таа точно работи. Типот на податокот што функцијата го враќа (повратна вредност) го определува типот на функцијата. Листата на параметри содржи имиња и типови на променливи преку кои функцијата добива вредности при нејзиното повикување

Функциите во C++ се декларираат со наредбата која има форма:
тип_на_функција име_на_функција(листа на параметри);

Оваа наредба се нарекува **прототип на функција**.

Пример. Во декларацијата:

`float pogolem (float a, float b);`

Името на функцијата е `pogolem`, типот на функцијата е `float` (резултатот е реален број), параметрите се променливите a и b , двете од типот `float` (реални променливи). **Имињата на**

параметрите може да се изостават, па функцијата `pogolem` може да се декларира и на следниов начин:

```
float pogolem (float, float);
```

Кога се декларира некоја функција, треба да се одговори на следниве прашања: **колку параметри (влезни податоци) има функцијата, од кој тип се тие податоци и каков тип ќе има резултатот (повратната вредност).**

На пр., функцијата `soberi` со која се собираат два реални броја, има два параметри и тие се од типот `float`. Резултатот на функцијата е збир на два реални броја, а тоа значи дека резултатот исто така ќе биде од типот `float`. Конечно, прототип на функцијата `soberi` ќе биде:

```
float soberi (float a, float b);
```

Основни математички функции – библиотека `cmath`

Во C++ постои стандардна математичка библиотека `cmath` која содржи многу математички функции. Најчесто користени се:

- `sqrt(x)` – квадратен корен од x
 - `exp(x)` – експоненцијална функција e^x
 - `log(x)` – природен логаритам од x (основа e)
 - `log10(x)` – декаден логаритам од x (основа 10)
 - `fabs(x)` – апсолутна вредност од x
 - `ceil(x)` – заокружување на x на најмалиот цел број не помал од x пр. 7.8 е 8 пр. 5.3 е 6
 - `floor(x)` – заокружување на x на најголемиот цел број не поголем од x (целиот дел од реален број) пр. 7.8 е 7 пр. 5.3 е 5
 - `round(x)` – заокружување на бројот пр. 7.8 е 8 пр. 5.3 е 5
 - `pow(x,y)` – x на степен y
 - `fmod(x,y)` – остаток од x/y како реален број пр. 7.8 со 3.2 е 1.4 пр. 5.3 со 4.2 е 1.1
 - `sin(x)` – синус од x
 - `cos(x)` – косинус од x
 - `tan(x)` – тангенс од x
- пр. Да се напише програма со која се пресметува и печати $s = \sqrt{1} + \sqrt{3} + \dots + \sqrt{n}$

```
# include <iostream>
# include <cmath>
using namespace std;
int main()
{
double s;
int i,n;
cin>>n;
s=0;
for (i=1;i<=n;i+=2)
s+=sqrt(i);
cout<<"zbirot na korenite e "<<s<<endl;
return 0;
}
```

пр. Да се напише програма со кој се пресметува и печати вредноста на `sin` за агол од 0 до 360 степени со чекор 10.

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
int alfa;
cout<<"alfa      sin(alfa)"<<endl;
cout<<"_____ "<<endl;
for (alfa=0;alfa<=360;alfa+=10)
    cout<<alfa<<"      "<< sin(alfa*3.14/180)<<endl;
return 0;
}

```

пр. Да се напише програма за пресметување на функцијата
 $f(x)=\cos(x)+4*\sin(x)-\tan(x)$

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
double x,y;
cout<<"vnesi ja vrednosta na x "<<endl;
cin>>x;
x1=x*3.14/180;
y=cos(x1)+4*sin(x1)-tan(x1);
cout<<"funkcijata e "<<y<<endl;
return 0;
}

```

пр. да се напише програма која со потпрограма пресметува збир на природните броеви од 1 до n.

```

#include <iostream>
using namespace std;

int soberi(int a)
{
    int s,i;
    s=0;
    for (i=1; i<=a; i++)
        s+=i;
    return s;
}

int main()
{
int a,n;
cout<<"vnesi go n ";
cin>>n;
cout<<"zbirot od 1 do "<<n<<" e "<<soberi(n)<<endl;
return 0;
}

```

пр. да се напише програма која со потпрограма пресметува збир на n природните броеви.

```

#include <iostream>
using namespace std;

int soberi(int a)
{
    int s,i,b;
    s=0;
    for (i=1; i<=a; i++)
    {

```

```

        cin>>b;
        s+=b;
    }
    return s;
}
int main()
{
    int a,n;
    cout<<"vnesi go n ";
    cin>>n;
    cout<<"zbirot "<<n<<" e "<<soberi(n)<<endl;
    return 0;
}

```

пр. да се напише програма која со потпрограма пресметува средна вредност на n природни броеви.

```

#include <iostream>
using namespace std;

double sredna(int a)
{
    int s,i,b;
    double sr;
    s=0;
    for (i=1; i<=a; i++)
    {
        cin>>b;
        s+=b;
    }
    sr=s*1.0/a;
    return sr;
}
int main()
{
    int a,n;
    cout<<"vnesi go n ";
    cin>>n;
    cout<<"srednata vrednost e "<<n<<" e "<<sredna(n)<<endl;

    return 0;
}

```

пр. да се напише програма која со потпрограма определува нзд со евклидов алгоритам.

```

#include <iostream>
using namespace std;

int nzd(int a, int b)
{
int p,ost;
if (a<b)
{
    p=a;
    a=b;
    b=p;
}
do
{
    ost=a%b;
    a=b;
    b=ost;
} while (b!=0);
return a;
}

int main()
{
int n,m;
cin>>n>>m;
cout<<" nzd e "<<nzd(n,m);

return 0;
}

```

пр. Дадена е плоштина на круг. Напиши програма со која ќе се одреди радиусот на кругот.

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
float p,r;
cout<<"Vnesi plostinata"<<endl;
cin>>p;
r=sqrt(p/3.14);
cout<<"radiusot e "<<r;
return 0;
}

```

пр. Дадена е плоштина на квадрат. Напиши програма со која ќе се одреди дијагоналата на квадратот.

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
float p,d,a;
cout<<"Vnesi ja plostinata"<<endl;
cin>>p;
a=sqrt(p);
d=a*sqrt(2);
}

```

```

    cout<<"dijagonalata e "<<d;
    return 0;
}

```

пр. Дадени се катетите a и b на правоаголен триаголник. Напиши програма со која ќе се одреди хипотенузата на триаголникот.

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    float a,b,c;
    cout<<"Vnesi gi katetite "<<endl;
    cin>>a>>b;
    c=sqrt(pow(a,2)+pow(b,2));
    cout<<"Hipotenuzata e "<<c;
    return 0;
}

```

пр. Дадени се страните на триаголник a , b и c . Напиши програма со која се пресметува плоштината на триаголникот по формулата на Херон:

$$P = \sqrt{s(s-a)(s-b)(s-c)}, \text{ каде } s = \frac{a+b+c}{2}$$

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    float a,b,c,s,p;
    cout << "Vnesi gi stranite na triagolnikot" << endl;
    cin>>a>>b>>c;
    s=(a+b+c)/2;
    p=sqrt(s*(s-a)*(s-b)*(s-c));
    cout<<"plostinata e "<<p;
    return 0;
}

```

пр. Напиши програма со која ќе се отпечатат квадратите и корените на броевите од 1 до 25 како што е прикажано:

број	квадрат	корен
1	1	1
2	4	1.41
...
25	625	5

```

#include <iostream>
#include <cmath>
using namespace std;

```

```

int main()
{
    int n,i;
    cout << "Vnesi go n" << endl;
    cin>>n;
    cout<<"broj   kvadrat   koren "<<endl;
    for(i=1; i<=n; i++)
        cout<<" "<<i<<"   "<<pow(i,2)<<"   "<<sqrt(i)<<endl;
    return 0;
}

```

```

    return 0;
}
int zbir(int n)
{
    int z=0;
    z=n%10+n/10;
    return z;
}

```