

Дәріс-6

C++ тіліндегі функция

Функцияға кіріспе

Функция – бұл белгілі бір тапсырманы орындау мақсатында программа ішіндегі *нұсқаулар тобы*.

Көптеген программалар үлкен тапсырмаларды орындайды, сондақтан оларды бірнеше ішкі тапсырмаларға бөлуге тура келеді. Сондықтан программистер әдетте өз программаларын *функциялар* деп аталатын кішкене қолайлы бөліктерге бөледі.

Функция – бұл белгілі бір тапсырманы орындау мақсатында программа ішіндегі нұсқаулар тобы. Үлкен программаны бір ұзақ нұсқаулық ретінде жазудың орнына, программист бірнеше кішігірім функцияларды жасайды, олардың әрқайсысы тапсырманың белгілі бір бөлігін орындайды. Содан кейін бұл кішігірім функцияларды жалпы тапсырманы орындау үшін қажетті ретпен орындауға болады.

Функцияларға негізделген программаның артықшылықтары

Қарапайым код. Бағдарлама коды функцияларға бөлінгенде, оны түсіну оңайырақ болады. Бірнеше кішігірім функцияларды оқу ұзақ программаға қарағанда әлдеқайда жеңіл.

Кодты қайта пайдалану. Функция программадағы кодтын қайталануын азайтады. Егер бағдарламадағы белгілі бір операция бірнеше жерде орындалса, онда бұл әрекетті орындау үшін функцияны бір рет жазып, қажет болған кезде орындауға болады. Функциялардың бұл артықшылығы кодты қайта пайдалану деп аталады.

Функцияларға негізделген программаның артықшылықтары

Қарапайым тестілеу. Программадағы әрбір тапсырма өз функциясында болған кезде тестілеу және жөндеу процестері жеңілдейді. Программисттер программадағы әр функцияны жеке тексеріп, оның тапсырманы дұрыс орындағанын анықтай алады. Бұл оқшаулар және қателерді түзету процесін жеңілдетеді.

Жылдам даму. Программист немесе программисттер тобы көптеген программалар жасайды делік. Олар әр программада бірнеше жалпы тапсырмаларды орындайды, мысалы, пайдаланушы аты мен парольді анықтау, ағымдағы уақытты шығару және т.б. Осы тапсырмалардың барлығына программалық кодты бірнеше рет жазудың мағынасы жоқ. Оның орнына, жиі кездесетін тапсырмалар үшін функциялар жазылады және бұл функциялар оларға қажет кез-келген программаға енгізілуі мүмкін.

Топтық жұмысты жеңілдету. Функциялар сонымен қатар программисттерге командаларда жұмыс істеуді жеңілдетеді. Программа әрқайсысы жеке тапсырманы орындайтын функциялар жиынтығы ретінде жасалынған кезде, бұл жағдайда әртүрлі программисттерге әртүрлі функцияларды жазу тапсырылуы мүмкін.

Функцияны анықтау және шақыру

Функция атауында жеткілікті сипаттама болуы керек, сондықтан сіздің кодты оқитын кез-келген адам функцияның не істейтінін нақты болжай алады. Си-да айнымалыларды атау кезіндегі сақталған ережелер функцияда да сақталуы керек:

- функция атауында бос орындар болмайды;
- бірінші таңба а-дан z-ге дейін, А-дан Z-ге дейін әріптердің бірі немесе астын сызу(_) символы болуы керек;
- бірінші таңбадан кейін а-дан z-ге дейін немесе А-дан Z-ге дейін әріптерді қолдануға болады, 0-ден 9-ға дейінгі сандар немесе астын сызу таңбалары;
- жоғарғы және төменгі әріптердегі таңбалар әр түрлі мағына береді.

C++ тіліндегі функция

Си-дегі кез-келген программа - бұл айнымалылар сипаттамасы мен функциялар жиынтығы.

Програмада кем дегенде, бір функция болады және программадағы бір функцияның аты main болуы тиіс. Программаның орындалуы осы main функциясынан басталады. Main функциясының орындалуы барысында басқа функциялар шақырылады

Функциялар арасындағы байланыс мән қайтаратын аргументтер мен сыртқы айнымалылар арқылы орнатылады. Return инструкциясы нәтижені шақыратын функциядан шақырған функцияға қайтару механизмін орындайды және return сөзінен кейін кез келген өрнек тұруы мүмкін.

```
return <өрнек>;
```

Өрнекті жақшаға да жазуға болады, бірақ ол міндетті емес.

Си-де функцианың барлық аргументі параметрлер ретінде беріледі, яғни шақырылатын функцияға аргументтің өзі емес оның аргументтерінің мәні беріледі. Си-гі кез келген функцияның прототипі болуы тиіс, яғни функция аты, аргументтерінің типі және қайтарылатын нәтиже типі көрсетілген тақырыбы хабарланады. Прототипі программа денесінде немесе бөлек тақырыптар файлында айқындалады. Функция прототипі компиляторға функция туралы информация береді де, сәйкесінше компилятор типтер сай келуін тексереді. Функция әртүрлі типті мән қайтара алады, int,void,float т.б.

Функция түрлері

Функция мынадай жалпы түрде сипатталады:

[<функция типі>] <функция аты> ([<формальды параметрлер спецификациясы>])
<функция денесі>;

мұндағы [<функция типі>] – функция есептейтін нәтиже типі;

<функцияның аты> – функция идентификаторы, яғни аты;

<формальды параметрлер спецификациясы> – бұл болмауы да мүмкін немесе әрбір параметр спецификациясы: <типі> <параметр аты> – формальды параметрлердің типтерін және идентификаторларын анықтайды;

<функция тұлғасы> – орындалатын операторлар тізбегі және оның ішінде локальді айнымалының сипатталуы болуы мүмкін.

Функцияны шақыру үшін функцияның аты және жақшада нақты аргументтері көрсетіледі. Функция өз жұмысын аяқтағаннан кейін есептелген нәтижені return операторы арқылы қайтарады.

Функцияларды хабарлау және анықтау

Функцияны хабарлау:

```
int sum (int a, int b);
```

Функцияны анықтау:

```
int sum (int a, int b)  
{  
    return (a+b);  
}
```


Функцияға мысал

```
#include <iostream>
int sum (int a, int b); //Функцияны хабарлау

int main(){
    int a = 2, b = 3, c, d;
    c = sum(a, b); //функцияны шақыру
    cin >> d;
    cout << sum(c, d); //функцияны шақыру
    return 0;
}
```

Функцияға мысал

```
#include <iostream>
Using namespace std;
void ekran();//прототип
{
cout<<"hello world\n";
}
int main()
{
    for(int i=0;i<10; i++)
        ekran();
}
```

Функцияға мысал

Оң бүтін санның факториалын анықтау функциясын былай жазуға болады:

```
long fakt(int k)  
{ if (k<0) return 0;  
  if (k==0) return 1;  
    long p=1;  
    for (int j=1; j<=k;j++)  
      p*=j;  
    return (p);  
}
```

Теріс сандардың факториалы болмайды, мұндайда функция 0 мәнін береді, ал $0!=1$, қалған жағдайларда бұл функция цикл ішінде факториалды есептейді.

Функцияға мысал

Мысалы: m, n бүтін сандары үшін мынадай өрнекті есептеу керек болсын делік:

*// $c = (m! * n!) / (m+n)!$ есептеу*

```
#include <conio.h>
#include <stdio.h>
float fak(int x)
{ float p=1;
  for (int j=1;j<=x;j++)
    p*=j;
  return (p);
}
main ()
{
  int m,n;
  float c;
  printf("введите m,n : ");
  scanf("%d%d",&m,&n);
  c=fak(m)*fak(n)/fak(m+n);
  printf(" %e\n",c);
  getch();
}
```

Функция

- Функция ішінде орналасқан айнымалылар типтері сипатталуы осы функцияның ішінде ғана анықталған болып саналады, (мұнда тек қана **main** функциясы ерекше жағдайда болып саналады). Программа мәтінінде функциялар кез келген ретпен орналасуы мүмкін. Функцияны шақыру үшін функцияның аты және жақшада нақты аргументтері көрсетіледі. Функция өз жұмысын аяқтағаннан кейін есептелген нәтижені **return** операторы арқылы қайтарады.