

Зертханалық жұмыс №8

Функциялар. Рекурсия

Жұмыс мақсаты: C++ тілінде функцияларды қолданудың практикалық дағдыларын алу.

8.1 Жалпы мәліметтер

Бас файлдарда орнатылған стандартты функциялардан басқа C++ тілі *пайдаланушылардың функцияларын* құрастыруға мүмкіндік береді. Егер де бірдей алгоритмдер бойынша есептеулер қайталанатын болса осындай функциялар жасалады. Функцияларды қолдану программаны қарапайым, жеңіл бақыланатын бөліктерге бөлуге мүмкіндік береді. Функцияларды қолданғанда функцияның бейнеленуін мен функцияны шақыратын операторларын айыру керек.

Функция құрылымы **main** программаның құрылымына ұқсас. Функция бейнеленуі функция басынан, айнымалыларды бейнелеуден және операторлардан тұрады:

```
функция_типі функция_аты (формалды_параметрлер_тізімі)  
{  
  айнымалыларды хабарлау;  
  Оператор;  
  ...;  
  Оператор;  
}
```

Мұндағы: *функция_типі* - негізгі программаға қайтарылатын нәтиженің типі; *функция_аты* – мағынасы бойынша функция орындайтын операцияға сәйкес уникалды ат; *формалды_параметрлер_тізімі* – формалды параметрлер тізімі және олардың типтері.

Функцияны шақыру үшін шақыратын программаның кез келген өрнегінде оның атын нақты параметрлер тізімімен көрсету жеткілікті:

```
функция_аты (нақты_параметрлер_тізімі);
```

Есептелген мәнді негізгі программаға қайтару үшін функцияларда келесі оператор қолданылады:

```
return (нәтиже);
```

Егер де функцияны шақыру оның бейнелуінен ерте орындалса, программа басына функцияның *прототипін* орнату керек. Функцияның прототипінде функцияның аты, қайталанатын мән типі, формалды параметрлер типтері мен саны туралы ақпарат орнатылады.

Жергілікті айнымалылар функцияның ішінде бас **main** функциядағы сияқты хабарланады. Жергілікті айнымалылар тек қана олар хабарланған функцияның ішінде жұмыс жасайды. *Глобалды айнымалылар* функциялардан тыс программа басында хабарланады. Программадағы кез келген функциядан глобалды айнымалыларды қолдануға болады. Егер де жергілікті және глобалды айнымалылардың аттары бірдей болса, функциядағы айнымалы C++ компиляторымен жергілікті айнымалы ретінде қабылданады. Егер де

функция ішінде жергілікті айнымалымен аты бірдей болатын глобальды айнымалыны қолдану қажет болса, келесі рұқсат беру глобальды операторды пайдалану керек:

:: айнымалы_аты

Глобальды айнымалылардың мәндерін кез келген функция өзгерте алатын болғандықтан, оларды программаларда қолданудан қашқақтаңыз.

8.2 Зертханалық жұмысқа тапсырма

Рекурсияны қолданып, 8.1 кестеден таңдалынған нұсқа бойынша есепті шешіңіз.

8.1 кесте – Тапсырма нұсқалары

Тапсырма	Нұсқа
1	
2	
3	
4	
5	
6	

8.1 кестенің жалғасы

7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	

8.1 кестенің соңы

25	
26	
27	
28	
29	
30	

8.3 Бақылау сұрақтары

- 8.3.1 Функцияның анықтамасын беріңіз.
- 8.3.2 Функцияны бейнелеу деген не?
- 8.3.3 Прототиптер қандай мақсатпен қолданылады?
- 8.3.4 Функция типінде *void* сөзі қандай мақсатпен қолданылады?
- 8.3.5 Функцияны қалай шақыруға болады?
- 8.3.6 Нақты параметр деген не?
- 8.3.7 Формалды параметрлердің нақты параметрлерден айырмашылығы не де?
- 8.3.8 Функциядан шығу қалай орындалады?
- 8.3.9 Рекурсивті функциялардың артықшылықтары мен кемшіліктерін атаңыз.
- 8.3.10 Жергілікті және глобалды параметрлер қандай мақсатпен қолданылады?

