

Процедуралар мен функциялар

Жеке программа түрінде бөлек жазылады, қажет кезінде оралуға болатын негізгі программаның арнайы бөлігі **қосалқы программа** деп аталады. Турбо Паскальда қайталанатын әрекеттерді қайта жазып көшіруге уақыт кетірмеу үшін қосалқы программалар ретінде функциялар мен процедураларды қолданады.

Паскаль тілінің барлық **процедуралары мен функциялары** екіге бөлінеді. Стандартты және тұтынушы өзі анықтаған бейстандарт топтар. Стандартты функциялар мен процедуралар тілдің элементі ретінде өз аттары бойынша алдын ала нақтыланған түйінді сөздермен белгіленеді. Ал бейстандарт функция мен процедураларды әрбір программалаушы өзі жаңа атау беру арқылы анықтайды да, олар орындалатын әрекеттерден тұрады.

Белгілі бір әрекеттерді орындау мақсатында алдын ала бекітілген аты бойынша шақырылып орындалатын программаның тәуелсіз бөлігі **процедура** деп аталады.

Жазылу ережесі.

*Procedure аты (көрсетпелі параметрлер тізімі);
сипаттау бөлігі*

Begin

Параметрлер бөлігі

end;

Жұмысының нәтижесі бір ғана мәнді анықтайтын оператор тізбегі **функция** деп аталады.

Жазылу ережесі.

function аты (формальді параметрлер тізімі): <нәтиже типі>;

Сипаттау бөлігі

Begin

Параметрлер бөлігі

end;

**Бейстандарт функциялар мен процедуралар
VAR бөлімінен кейін сипатталады.
әрбір программа соныда қойылады. ;**

**Функцияның Процедурадан екі
айырмашылығы бар.**

- 1.Функция аты өрнектердің аргумент ролін
атқара алады;**
- 2 .функция орындалған соң программаға
тек бір ғана мән беріледі.**

Процедуралар

Ерекшеліктері:

- әрбір формалды параметрдің қос нүктеден кейін типі көрсетілуі қажет

```
procedure A (x: real; y: integer; z: real);
```

- егер типтері бірдей параметрлер бірінен кейін бірі орналасса, олардың арасы үтірмен ажыратылады

```
procedure A (x, z: real; y, k, l: integer);
```

- процедураның ішіндегі параметрлер айнымалылар секілді қолданылады

Процедуралар

Ерекшеліктері:

- процедурада қосымша айнымалы ретінде жергілікті (локальді) айнымалыларды пайдалануға болады, басқа процедуралар оған қатынай алмайды

```
program qq;
```

```
  procedure A(x, y: integer);
```

```
    var a, b: real;
```

```
    begin
```

```
      a := (x + y) / 6;
```

```
      ...
```

```
    end;
```

```
begin
```

```
  ...
```

```
end.
```

локальді
айнымалылар

Функциялар

Ерекшеліктері:

- тақырыбы **function** сөзімен басталады

```
function Max (a, b: integer): integer;
```

- функцияда формальды параметрлер процедурадағыдай сипатталады

```
function qq( a, b: integer; x: real ): real;
```

- айнымалы параметрлерді пайдалануға болады

```
function Max (var a, b: integer): integer;
```

- тақырыптың соңында қоспуктеден кейін **нәтиженің типі** көрсетіледі

- функция

```
function Max (a, b: integer): integer;
```

 аласады

Функциялар

Есеп: екі санның үлкенін анықтайтын функция құру және оны қолдануға мысал келтіру

Функция:

формальды параметрлер

```
function Max (a, b: integer): integer;  
begin  
  if a > b then Max := a  
  else       Max := b;  
end.
```

бұл функцияны нәтижесі

Программа

```
program qq;  
var a, b, c: integer;  
  
function Max (a, b: integer): integer;  
begin  
    ...  
end;  
  
begin  
    writeln('Екі сан енгіз');  
    read(a, b);  
    c := Max ( a, b );  
    writeln('Санның Үлкені ', max c  
end.
```

нақты параметрлер

функцияны шақыру



Айнымалылардың, функциялардың және процедуралардың аттары бірдей болмауы керек!