Лабораторна робота №6

Процедури та функції користувача

Мета роботи: здобуття практичних навичок створення та використання процедур та функцій користувача.

Теоретичні відомості

Pascal передбачає створення структурованих програм з чіткою і добре розвиненою ієрархією. Для виконання частин програми, які логічно можуть бути виділені в окремі блоки, використовуються підпрограми. Підпрограма - частина програми, оформлена у вигляді окремої синтаксичної конструкції, що може включати практично усі риси, властиві програмі в цілому. Підпрограма завжди має ім'я, яке служить для виклику підпрограми з програми або з іншої підпрограми.

Характерним для організації підпрограм є те, що вони можуть мати набір власних локальних об'єктів: констант, типів, змінних та ін. які доступні тільки всередині цієї підпрограми, на відміну від глобальних об'єктів програми, які доступні в будь-якому місці програми, включаючи підпрограми.

TurboPascal дозволяє створювати підпрограми двох основних типів: procedure та function.

6.1. Підпрограма `Procedure`

Підпрограма типу **`Procedure`** в найпростішому випадку має синтаксис:

`Procedure <PrName> [(<param.list>)];`

де **`<PrName>`** ідентифікатор процедури (ім'я);

`<param.list>` список параметрів процедури;

До списку параметрів вноситься опис параметрів, через які **`Procedure`** спілкується з блоком вищого рівня ієрархії, з якого вона викликана (основною програмиою чи підпрограмою).

Приклад:

```
procedure Swap(var: a1,a2 :real);
var  atmp: real;
Begin
     atmp:=a1;
     a1:=a2;
     a2:=atmp;
End;
```

Процедура проводить обмін значеннями двох вказаних при виклику змінних. Виклик процедури рядком **Swap(X[i], X[j])**; призведе до обміну данними двох елементів масиву X.

З точки зору взаємодії з процедурою можна виділити чотири класи параметрів:

- глобальні - параметри, що описані в блоці вищого рівня ієрархії; використання таких параметрів в процедурі без використання механізму передачі не рекомендується.
- формальні по значенню - локальні параметри, описані в заголовку процедури, і призначені для передачі даних в процедуру; значення таких параметрів не повинно змінюватись при роботі підпрограми.
- формальні змінні - локальні параметри, описані в заголовку процедури, і призначені для двосторонньої передачі даних в процедуру та з процедури в програму; при описуванні таких параметрів перед ними ставиться слово **var**.
- прості локальні параметри - параметри, описані беспосередньо в процедурі; їх значення недоступне для безпосередньої зміни ззовні підпрограми.

Усі параметри, крім глобальних, існують тільки на час роботи процедури. Після виходу з процедури ці змінні зникають (звільняють пам'ять) і з'являються тільки у випадку повторного звертання до процедури.

6.2. Функції Function

Підпрограма типу Function має синтаксис:

```
Function <FnName> [(<param.list>)]: <FnType>;
```

де **<FnName>** ідентифікатор функції (ім'я);

<param.list> список параметрів функції;

<FnType> тип результату, який повертає функція.

Основні відмінності підпрограми типу **Function** від підпрограми типу **Procedure** такі:

- **Function**, звичайно, не викликається окремим командним рядком, а виступає як параметр в операторах присвоювання, інших функціях та процедурах;
- У структурі функції повинен бути оператор присвоювання (звичайно останній в тілі функції), у лівій частині якого знаходиться змінна, що співпадає з ідентифікатором функції, при цьому змінна додатково не описується.

Приклад:

```
Function Sign(s_a:real):shortint;
Begin
     if s_a<0 then Sign:=-1 else Sign:=1;
     end;

Function Power(p_a:real, p_b:integer):real;
var PowTmp:real;
Begin
     PowTmp:=exp(p_b * ln(abs(p_a)));
     if Odd(p_b) then Power := Sign(p_a) * PowTmp
          else Power := PowTmp;
     end;
```

Наведені функції дозволяють підняти будь-яке дійсне число до будь-якої цілої степені. Для довідки: стандартна функція **Odd(X:Longint):Boolean;** повертає значення **TRUE**, якщо X - непарне число.

У програмі звернення до функції може мати такий вигляд:

```
For i:=1 to N do
S := S + Power(A[i],i);
```

Це дозволить одержати суму елементів масиву в степені, що рівна номеру елемента масиву.

Завдання на лабораторну роботу.

Скласти та відлагодити програму за вказівкою викладача, в якій будуть використовуватись процедури та функції користувача

Рекомендована література: [1], [2], [4], [5], [6] , [7], [8]