**TIN HỌC 11**

**ÔN TẬP HỌC KỲ II (tiếp theo)**

**CHƯƠNG VI: CHƯƠNG TRÌNH CON VÀ LẬP TRÌNH CÓ CẤU TRÚC**

**BÀI 17: CHƯƠNG TRÌNH CON VÀ PHÂN LOẠI**

**1. Khái niệm chương trình con**

Chương trình con là một dãy lệnh mô tả một số thao tác nhất định và có thể được thực hiện (được gọi) từ nhiều vị trí trong chương trình.

**\* Lợi ích của việc sử dụng chương trình con:**

- Tránh được việc phải viết lặp đi lặp lại cùng một dãy lệnh nào đó.

- Hỗ trợ việc thực hiện các chương trình lớn.

- Phục vụ cho quá trình trừu tượng hóa.

- Mở rộng khả năng ngôn ngữ.

- Thuận tiện cho phát triển, nâng cấp chương trình.

**2. Phân loại và cấu trúc của chương trình con**

**a) Phân loại**

- Chương trình con gồm hai loại:

* **Hàm (function)** là chương trình con thực hiện một số thao tác nào đó và trả về một giá trị qua tên của nó. Ví dụ hàm toán học hay hàm xử lí xâu:

**sin(x)** nhận giá trị thực x và trả về giá trị sinx,

**length(x)** nhận xâu x và trả về độ dài của xâu x,...

* **Thủ tục (procedure):** là chương trình con thực hiện các thao tác nhất định nhưng không trả về giá trị nào qua tên của nó. Ví dụ các thủ tục vào/ra chuẩn hay thủ tục xử lí xâu:

**writeln, readln, delete, insert,...**

**b) Cấu trúc chương trình con**

Chương trình con có cấu trúc tương tự chương trình, nhưng nhất thiết phải có tên và phần đầu dùng để khai báo tên, nếu là hàm phải khai báo kiểu dữ liệu cho giá trị trả về của hàm:

<phần đầu>

[<phần khai báo>]

<phần thân>

**Phần khai báo**

Phần khai báo có thể có khai báo biến cho dữ liệu vào và ra, các hằng và biến dùng trong chương trình con.

**Phần thân**

Phần thân của chương trình con là dãy câu lệnh thực hiện để từ những dữ liệu vào ta nhận được dữ liệu ra hay kết quả mong muốn.

* Tham số hình thức
* Các biến được khai báo cho dữ liệu vào/ra được gọi là *tham số hình thức* của chương trình con.
* Các biến được khai báo để dùng riêng trong chương trình con được gọi là *biến cục bộ*. Chương trình chính và các chương trình con khác không sử dụng được biến cục bộ này.
* Các biến được khai báo trong chương trình chính gọi là *biến toàn cục*. Mọi chương trình con đều sử dụng được biến toàn cục của chương trình chính.

**c) Thực hiện chương trình con**

* **Tham số thực sự**

Để thực hiện (gọi) một chương trình con, ta cần phải có lệnh gọi nó tương tự lệnh gọi hàm hay thủ tục chuẩn, bao gồm tên chương trình con với tham số (nếu có) là các hằng và biến chứa dữ liệu vào và ra tương ứng với các tham số hình thức đặt trong cặp ngoặc ( và ). Các hằng và biến này được gọi là các tham số thực sự. Ví dụ: sqr(25).

Khi thực hiện chương trình con, các tham số hình thức để nhập dữ liệu vào sẽ nhận giá trị của tham số thực sự tương ứng, còn các tham số hình thức để lưu trữ dữ liệu ra sẽ trả giá trị đó cho tham số thực sự tương ứng.

Ví dụ, khi thực hiện tính *TLuythua* cần bốn lần gọi chương trình con *Luythua(x, k)* với các tham số *(a, n)*, *(b, m)*, *(c, p)*, *(d, q)* và các tham số này là tham số thực sự tương ứng với tham số hình thức *(x, k)*.

**BÀI 18: VÍ DỤ VỀ CÁCH VIẾT VÀ SỬ DỤNG CHƯƠNG TRÌNH CON**

**1. Cách viết và sử dụng thủ tục:**

**a. Cấu trúc của thủ tục:**

**procedure** <Tên thủ tục>[(<danh sách tham số>)];

[<phần khai báo>]

**begin**

[<dãy các lệnh>]

**end;**

**b. Ví dụ viết thủ tục tên trường (Ten\_Truong) không có tham số để vẽ chữ TVG bằng các dấu \* trên màn hình máy tính.**

procedure Ten\_Truong;

begin

writeln('\*\*\*\*\*  \*     \*  \*\*\*\*\*');

writeln('    \*       \*   \*   \*       ');

writeln('    \*        \* \*    \* \*\*\*');

writeln('    \*         \*      \*\*\*\*\*');

end;

**c. Tham trị và tham biến**

- Trong lệnh gọi thủ tục, các tham số hình thức được thay bằng các tham số thực sự tương ứng là các giá trị cụ thể được gọi là các **tham số giá trị** (gọi tắt là *tham trị*).

- Trong lệnh gọi thủ tục, các tham số hình thức được thay bằng các tham số thực sự tương ứng là tên các biến chứa dữ liệu ra được gọi là các **tham số biến** (gọi tắt là *tham biến*).

- Để phân biệt tham biến và tham trị, Pascal sử dụng từ khoá **var** để khai báo những tham số biến.

**2. Cách viết và sử dụng hàm:**

**\* Chú ý:** Điểm khác nhau cơ bản giữa thủ tục và hàm là việc thực hiện hàm luôn trả về giá trị kết quả thuộc kiểu xác định và giá trị đó được gán cho tên hàm.

**a. Cấu trúc của hàm:**

**function** <Tên hàm>[(<danh sách tham số>)]:<kiểu dữ liệu>;

[<phần khai báo>]

**begin**

[<dãy các lệnh>]

**end;**

- Trong đó *kiểu dữ liệu* là kiểu dữ liệu của giá trị mà hàm trả về và chỉ có thể là các kiểu *integer, real, char, boolean, string*.

- **Trong thân hàm bắt buộc phải có câu lệnh gán giá trị cho tên hàm:**

<tên hàm>:= <biểu thức>;

**b. Ví dụ viết hàm tìm giá trị lớn nhất (Max) của hai số a và b.**

Function Max(a,b:real):real;

Begin

if a>b then Max:=a else Max:=b;

End;

**3. Cách khai báo và sử dụng chương trình con khi viết chương trình Pascal**

- Các chương trình con (thủ tục hoặc hàm) được khai báo trong phần khai báo của chương trình chính, ngay sau phần khai báo biến.

- Khi cần thực hiện ta viết lệnh gọi chương trình con trong chương trình chính theo cú pháp **<Tên chương trình con>[<danh sách tham số>];** trong đó danh sách tham số có thể có hoặc không.

**4. Ví dụ chương trình con**

Viết chương trình sử dụng các chương trình con đọc 2 số a và b từ tệp INPUT.TXT và ghi tổng (S) của 2 số a, b đó vào tệp OUTPUT.TXT. **(Chú ý phần đặt giữa cặp dấu { và } là chú thích trong Pascal)**

**\* Đầu tiên ta thử viết chương trình bình thường không sử dụng chương trình con.**

program ViDu;

var a, b, S: real;

 f1,f2: text;

BEGIN

 {Chu thich: Qua trinh doc file INPUT.TXT}

 assign(f1,'INPUT.TXT');

 reset(f1);

 read(f1,a,b);

 close(f1);

 {Chu thich: Qua trinh ghi file OUTPUT.TXT}

 assign(f2,'OUTPUT.TXT');

 rewrite(f2);

 S:=a+b;

 write(f2,S);

 close(f2);

END.

**\* Sau đó ta viết chương trình sử dụng các chương trình con là thủ tục Nhap (dùng để đọc file INPUT.TXT), hàm Tong (dùng để tính tổng a và b), thủ tục Xuat (dùng để ghi file OUTPUT.TXT). Chương trình thu được như sau:**

program ChuongTrinhCon;

var a, b, S: real;

 f1,f2: text;

{Khai bao thu tuc Nhap}

procedure Nhap;

begin

 {Chu thich: Qua trinh doc file INPUT.TXT}

 assign(f1,'INPUT.TXT');

 reset(f1);

 read(f1,a,b);

 close(f1);

end;

{Khai bao ham Tong}

function Tong(a,b:real):real;

begin

 Tong:=a+b;

end;

{Khai bao thu tuc Xuat}

procedure Xuat;

begin

 {Chu thich: Qua trinh ghi file OUTPUT.TXT}

 assign(f2,'OUTPUT.TXT');

 rewrite(f2);

 S:=Tong(a,b);{Tong S duoc gan bang gia tri cua ham Tong}

 write(f2,S);

 close(f2);

end;

BEGIN

 Nhap;{Goi thu tuc Nhap}

 Xuat;{Goi thu tuc Xuat}

END.

**5. Bài tập ôn tập chương trình con**

Học sinh lên trang ViettelStudy làm các bài tập ôn tập theo yêu cầu của giáo viên.