**Bài 15: THAO TÁC VỚI TỆP**

**1. Khai báo biến tệp**

Var <tên biến tệp>:Text;

- Ví dụ: Khai báo xác định 2 biến tệp có tên là f1, f2

Var F1, F2: TEXT;

- Độ dài lớn nhất của tên tệp là 79 ký tự.

**2. Thao tác với tệp** :

***a. Gán tên tệp***

ASSIGN (<Tên biến tệp>,<Tên tệp>);

Ví dụ : **ASSIGN** (F,’DULIEU.DAT’);

+ Tên tệp có đường dẫn, tệp lưu ở thư mục chỉ ra trong đường dẫn

Ví dụ: ASSIGN (tep1,’**D:\TP**\UCLN.bak’);

+ **Tên tệp không có đường dẫn**

Ví dụ: ASSIGN (tep2,’KQ.txt’);

***b. Mở tệp***

\*  **T**hủ tục mở tệp để ghi:

**REWRITE** (<Tên biến tệp>;

Ví dụ:

Assign (F,’DULIEU.DAT’);

Rewrite(F);

* Tệp chưa có sẵn tệp thì tệp đó🡪 tạo tệp với nội dung rỗng
* Tệp đó đã tồn tại 🡪 nội dung cũ sẽ bị xóa.

\* Thủ tục mở tệp để đọc

**RESET(<**Tên biến tệp**>);**

* + Mở một tệp đã tồn tại
  + Dữ liệu cũ của tệp không bị xóa

Ví dụ: Assign(F3, ’DULIEU.DAT’);

Reset(F3);

**c. Đọc ghi kiểu dữ liệu:**

+ Câu lệnh dùng thủ tục đ**ọc tệp**

**READ** (<Biến tệp>, <Danh sách biến>); hoặc

**READLN**(<Biến tệp>, <Danh sách biến>);

Readln(f,x,y);

+ Câu lệnh dùng thủ tục **ghi tệp**

**WRITE (**<Tên biến tệp >, <Danh sách kết quả>);hoặc

**WRITELN (**<Tên biến tệp >, <Danh sách kết quả>);

**d. Đóng tệp**:

CLOSE(<tên biến tệp>);

**e. Một số hàm và thủ tục** chuẩn dùng trong thao tác tệp:

- EOF(<biến tệp>): cho giá trị TRUE nếu con trỏ tệp đang chỉ tới cuối tệp.

- EOLN(<biến tệp>): trả về giá trị TRUE nếu con trỏ tệp đang chỉ tới cuối dòng

d. Ví dụ ứng dụng:

+ ví dụ 1: **Viết chương trình** đọc hai biến a, b kiểu số nguyên vào **tệp** ‘**Dulieu.txt**’

**Var**

tep1: TEXT;

a,b: integer;

**Begin**

assign (tep1, ‘dulieu.txt’);

reset (tep1);

read (tep1, a, b);

**Close** (tep1); {Đóng tệp}

**End.**

Ví dụ 2: **Viết chương trình ghi** hai biến a, b kiểu số nguyên **vào tệp** ‘**Dulieu.txt**’. Cho biết a có giá trị 2, b có giá trị 5 viết cách nhau có khoảng trắng

**Var** tep2: TEXT;

a,b: integer;

**Begin**

assign (tep2, ‘dulieu.txt’);

rewrite (tep2); {Mở tệp để ghi}

a:= 2; b:= 5;

write ( tep2, a**,’ ‘,** b);

**Close** (tep2); {Đóng tệp}

**End.**

**Bài 16: VÍ DỤ LÀM VIỆC VỚI TỆP**

1. Ví dụ 1 (Yêu cầu đề bài SGK, trang 87)

Progam Khoang\_cach;

Var d: real; f :text; x, y: integer;

Begin

Assign(f, ‘TRAI.TXT’);

Reset(f);

While not eof(f) do

Begin

Read(f,x,y);

D:= sqrt(x\*x+y\*y);

Write (‘Khoang cach: ‘,d:10:2);

End;

Close(f);

End.

2. Ví dụ 2 (SGK, trang 87)

Program Dien\_tro;

Var a: array[1..5] of real;

R1, R2, R3: real; f1,f2 :text; i: integer;

Begin

Assign(f1, ‘RESIST.DAT’);

Reset(f1);

Assign(f2, ‘RESIST.EQU’);

Rewrite(f2);

While not eof(f1) do

Begin

Readln(f1,R1,R2,R3);

a[1]:=R1\*R2\*R3/(R1\*R2+R1\*R3+R2\*R3);

a[2]:=R1\*R2/(R1+R2)+R3;

a[3]:=R1\*R3/(R1+R3)+R2;

a[4]:=R2\*R3/(R2+R3)+R1;

a[5]:=R1+R2+R3;

For i:=1 to 5 do write(f2, a[i]:9:3,’ ‘);

Writeln(f2);

End;

Close(f1); Close(f2);

End.

**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 từ nhiều vị trí trong chương trình.*

**\* Lợi ích của việc sử dụng CTC**

+ Tránh được việc phải viết đi viết lại nhiều lần cùng 1 dòng lệnh;

+ Hổ trợ việc thực hiện các chương trình lớn;

+ Phục vụ cho quá trình trừu tượng hoá;

+ 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**:

* Hàm (**Function**) là chương trình con thực hiện một số thao tác nào đó và **trả về một giá trị** qua tên của nó.
* Thủ tục (**Procedure**) là chương trình con thực hiện một số thao tác nào đó nhưng **không trả về một giá trị** nào qua tên của nó.

2. Phân loại và cấu trúc chương trình con:

**b. Cấu trúc chương trinh con**

**<Phần đầu>**

**[<Phần khai báo >]**

**<Phần thân >**

**+Phần đầu:**

+ Để khai báo tên của hàm hoặc thủ tục.

+ Nếu là hàm phải khai báo kiểu dữ liệu chi giá trị trả về của hàm.

+ Nhất thiết phải có.

**+Phần khai báo:**

* Khai báo các biến cho dữ liệu vào/ra, các hằng và biến dùng trong chương trình con.

**+Phần thân:**

* Gồm dãy các lệnh thực hiện để từ những dữ liệu vào/ra ta nhận dữ liệu ra hay kết qủa mong muốn.

***\*Khái niệm các biến:***

*- Tham số hình thức*: gồm các biến được khai báo cho dữ liệu vào/ra.

*- Biến cục bộ:* Gồm các biến khái được khai báo trong chương trình con.

*- Biến toàn cục:* Gồm các biến khái được khai báo trong chương trình chính.

**\*Phạm vi hoạt động của các biến:**

-*Biến cục bộ:*

* Chỉ sử dụng trong một chương trình con của nó mà thôi.
* Không thể sử dụng biến cục bộ cuả một chương trình con cho chương trình chính và các chương trình con khác.

*-Biến toàn cục:*

* Được sử dụng trong chương trình chính cũng có thể sử dụng trong chương trình con.

**c. Thực hiện chương trình con:**

-Để thực hiện gọi chương trình con ta thực hiện lệnh theo có pháp sau

**Cú pháp :**

<tên chương trình con>(<tham số thực sự>)

*Trong đó*: tham số thực sự là các hằng, biến chứa dữ liệu vào/ ra.

Ví dụ:

Sqr(225) thì sqr là tên chương trình còn 225 là tham số thực sự