**GIỚI THIỆU TÓM TẮT TẬP TÀI LIỆU**

**1. Sự cần thiết:**

Các trường THCS đang có xu hướng dạy môn tự chọn tin học thay cho chủ đề tự chọn khi mà cơ sở vật chất phục vụ cho việc dạy học môn học này được trang bị ngày càng tốt hơn.

**2. Nội dung:**

Tập tài liệu được biên soạn theo định hướng 10 x 10. Nội dung bồi dưỡng được chia làm 10 chương. Mỗi chương gồm tóm tắt lý thuyết và 10 bài tập xoay quanh nội dung của chương. Mỗi bài tập được trình bày theo cấu trúc:

a. Đề bài.

b. Hướng dẫn, thuật toán.

c. Mã chương trình.

d. Nhận xét: Nhấn mạnh nội dung mới, quan trọng cần nắm sau khi thực hiện bài tập, giải quyết bài toán theo thuật toán khác, điểm được và chưa được của thuật toán ...

Nội dung các bài tập ở các chương đã được lựa chọn theo hướng kế thừa, tăng dần độ khó. Nhiều bài toán cần giải quyết trong thực tế được đưa vào các bài tập nhằm tăng hướng thú học tập ....

Hầu hết các bài tập có độ khó vừa phải, phù hợp với nội dung bồi dưỡng ở cấp trường. Nội dung liên quan với môn toán và các môn khác từ lớp 8 trở xuống. Nội dung bồi dưỡng được chia thành 10 chương như sau:

**Lớp 9**:

I. Làm quen với chương trình Pascal – Khai báo, sử dụng biến – Các thủ tục vào ra.

II. Cấu trúc lựa chọn: if … then … else

Case ... of ...

III. Cấu trúc lặp với số lần lặp đã biết: For … to … do

IV. Cấu trúc lặp với số lần lặp chưa biết.

V. Dữ liệu kiểu mảng (một chiều).

VI. Chương trình con.

VII. Chuyên đề: Tính chia hết- Số nguyên tố.

VIII. Chuyên đề dãy con.

IX. Chuyên đề chữ số - hệ cơ số.

X. Chuyên đề đa thức.

**3. Đề nghị:**

Chắc chắn tập tài liệu cần hiệu chỉnh, bổ sung để có thể đưa vào sử dụng. Rất mong Hội đồng thẩm định cho y kiến cụ thể về:

- Những hiệu chỉnh về cấu trúc của tập tài liệu.

- Những nội dung cần bổ sung thêm, những nội dung cần giảm bớt cho phù hợp với thực tế bồi dưỡng tại các đơn vị trường.

- Đề xuất các bài tập hay hơn để thay thế cho các bài tập đã có ở tập tài liệu. Các bài tập mà tập tài liệu còn thiếu.

CHƯƠNG I

CÁC KIỂU DỮ LIỆU CƠ BẢN

KHAI BÁO HẰNG, BIẾN, KIỂU, BIỂU THỨC VÀ CÂU LỆNH

A. LÝ THUYẾT:

I. CÁC KIỂU DỮ LIỆU CƠ BẢN

1. Kiểu logic

- Từ khóa: **BOOLEAN**

- miền giá trị: **(TRUE, FALSE)**.

- Các phép toán: phép so sánh (=, <, >) và các phép toán logic: AND, OR, XOR, NOT.

Trong Pascal, khi so sánh các giá trị boolean ta tuân theo qui tắc: FALSE < TRUE.

Giả sử A và B là hai giá trị kiểu Boolean. Kết quả của các phép toán được thể hiện qua bảng dưới đây:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** | **A AND B** | **A OR B** | **A XOR B** | **NOT A** |
| TRUE | TRUE | TRUE | TRUE | FALSE | FALSE |
| TRUE | FALSE | FALSE | TRUE | TRUE | FALSE |
| FALSE | TRUE | FALSE | TRUE | TRUE | TRUE |
| FALSE | FALSE | FALSE | FALSE | FALSE | TRUE |

**2. Kiểu số nguyên**

***2.1. Các kiểu số nguyên***

|  |  |  |
| --- | --- | --- |
| **Tên kiểu** | **Phạm vi** | **Dung lượng** |
| Shortint | -128 → 127 | 1 byte |
| Byte | 0 → 255 | 1 byte |
| Integer | -32768 → 32767 | 2 byte |
| Word | 0 → 65535 | 2 byte |
| LongInt | -2147483648 → 2147483647 | 4 byte |

***2.2. Các phép toán trên kiểu số nguyên***

*2.2.1. Các phép toán số học:*

**+, -, \*, /** (phép chia cho ra kết quả là số thực).

Phép chia lấy phần nguyên: **DIV** (Ví dụ : 34 DIV 5 = 6).

Phép chia lấy số dư: **MOD** (Ví dụ: 34 MOD 5 = 4).

**3. Kiểu số thực**

***3.1. Các kiểu số thực:***

|  |  |  |
| --- | --- | --- |
| **Tên kiểu** | **Phạm vi** | **Dung lượng** |
| Single | 1.5×10-45→ 3.4×10+38 | 4 byte |
| Real | 2.9×10-39→ 1.7×10+38 | 6 byte |
| Double | 5.0×10-324→ 1.7×10+308 | 8 byte |
| Extended | 3.4×10-4932→ 1.1×10+4932 | 10 byte |

***Chú ý***: Các kiểu số thực Single, Double và Extended yêu cầu phải sử dụng chung với bộ đồng xử lý số hoặc phải biên dich chương trình với chỉ thị **{$N+}** để liên kết bộ giả lập số.

***3.2. Các phép toán trên kiểu số thực:* +, -, \*, /**

***Chú ý***: *Trên kiểu số thực không tồn tại các phép toán DIV và MOD.*

***3.3. Các hàm số học sử dụng cho kiểu số nguyên và số thực:***

SQR(x): Trả về x2

SQRT(x): Trả về căn bậc hai của x (x≥0)

ABS(x): Trả về |x|

SIN(x): Trả về sin(x) theo radian

COS(x): Trả về cos(x) theo radian

ARCTAN(x): Trả về arctang(x) theo radian

TRUNC(x): Trả về số nguyên gần với x nhất nhưng bé hơn x.

INT(x): Trả về phần nguyên của x

FRAC(x): Trả về phần thập phân của x

ROUND(x): Làm tròn số nguyên x

PRED(n): Trả về giá trị đứng trước n

SUCC(n): Trả về giá trị đứng sau n

ODD(n): Cho giá trị TRUE nếu n là số lẻ.

INC(n): Tăng n thêm 1 đơn vị (n:=n+1).

DEC(n): Giảm n đi 1 đơn vị (n:=n-1).

**4. Kiểu ký tự**

- Từ khoá: **CHAR**.

- Kích thước: 1 byte.

- Để biểu diễn một ký tự, ta có thể sử dụng một trong số các cách sau đây:

* Đặt ký tự trong cặp dấu nháy đơn. Ví dụ 'A', '0'.
* Dùng hàm CHR(n) (trong đó n là mã ASCII của ký tự cần biểu diễn). Ví dụ CHR(65) biễu diễn ký tự 'A'.
* Dùng ký hiệu #n (trong đó n là mã ASCII của ký tự cần biểu diễn). Ví dụ #65.

- Các phép toán: =, >, >=, <, <=,<>.

**\* Các hàm trên kiểu ký tự:**

- **UPCASE(ch)**: Trả về ký tự in hoa tương ứng với ký tự ch. Ví dụ: UPCASE('a') = 'A'.

- **ORD(ch)**: Trả về số thứ tự trong bảng mã ASCII của ký tự ch. Ví dụ ORD('A')=65.

- **CHR(n)**: Trả về ký tự tương ứng trong bảng mã ASCII có số thứ tự là n. Ví dụ: CHR(65)='A'.

- **PRED(ch)**: cho ký tự đứng trước ký tự ch. Ví dụ: PRED('B')='A'.

- **SUCC(ch)**: cho ký tự đứng sau ký tự ch. Ví dụ: SUCC('A')='B'.

**II. KHAI BÁO HẰNG**

- Hằng là một đại lượng có giá trị không thay đổi trong suốt chương trình.

- Cú pháp:

**CONST <Tên hằng> = <Giá trị>;**

**III. KHAI BÁO BIẾN**

- Biến là một đại lượng mà giá trị của nó có thể thay đổi trong quá trình thực hiện chương trình.

- Cú pháp:

**VAR <Tên biến>[,<Tên biến 2>,...] : <Kiểu dữ liệu>;**

Ví dụ:

VAR x, y: Real; {Khai báo hai biến x, y có kiểu là Real}

a, b: Integer; {Khai báo hai biến a, b có kiểu integer}

***Chú ý***: *Ta có thể vừa khai báo biến, vừa gán giá trị khởi đầu cho biến bằng cách sử dụng cú pháp như sau*:

**CONST <Tên biến>: <Kiểu> = <Giá trị>;**

Ví dụ:

CONST x:integer = 5;

Với khai báo biến x như trên, trong chương trình giá trị của biến x có thể thay đổi. (Điều này không đúng nếu chúng ta khai báo x là hằng).

**IV. BIỂU THỨC**

Biểu thức (expression) là công thức tính toán mà trong đó bao gồm các phép toán, các hằng, các biến, các hàm và các dấu ngoặc đơn.

Ví dụ: (x +y)/(5-2\*x) biểu thức số học

(x+4)\*2 = (8+y) biểu thức logic

Trong một biểu thức, thứ tự ưu tiên của các phép toán được liệt kê theo thứ tự sau:

* Lời gọi hàm.
* Dấu ngoặc ()
* Phép toán một ngôi (NOT, -).
* Phép toán \*, /, DIV, MOD, AND.
* Phép toán +, -, OR, XOR
* Phép toán so sánh =, <, >, <=, >=, <>, IN

**V. CÂU LỆNH**

**6.1. Câu lệnh đơn giản**

- ***Câu lệnh gán (:=)***: <Tên biến>**:=**<Biểu thức>;

- Các lệnh xuất nhập dữ liệu: **READ/READLN, WRITE/WRITELN.**

- Lời gọi hàm, thủ tục.

**6.2. Câu lệnh có cấu trúc**

- Câu lệnh ghép: **BEGIN ... END;**

- Các cấu trúc điều khiển: **IF.., CASE..., FOR..., REPEAT..., WHILE...**

**6.3. Các lệnh xuất nhập dữ liệu**

***6.3.1. Lệnh xuất dữ liệu***

Để xuất dữ liệu ra màn hình, ta sử dụng ba dạng sau:

**(1) WRITE(<tham số 1> [, <tham số 2>,...]);**

**(2) WRITELN(<tham số 1> [, <tham số 2>,...]);**

**(3) WRITELN;**

Các thủ tục trên có chức năng như sau:

1. Sau khi xuất giá trị của các tham số ra màn hình thì con trỏ không xuống dòng.
2. Sau khi xuất giá trị của các tham số ra màn hình thì con trỏ xuống đầu dòng tiếp theo.
3. Xuống dòng.

Các tham số có thể là các hằng, biến, biểu thức. Nếu có nhiều tham số trong câu lệnh thì các tham số phải được phân cách nhau bởi dấu phẩy.

Khi sử dụng lệnh WRITE/WRITELN, ta có hai cách viết: ***không qui cách*** và ***có qui cách***:

- ***Viết không qui cách***: dữ liệu xuất ra sẽ được canh lề ở phía bên trái. Nếu dữ liệu là số thực thì sẽ được in ra dưới dạng biểu diễn khoa học.

Ví dụ:

WRITELN(x); WRITE(sin(3\*x));

- ***Viết có qui cách***: dữ liệu xuất ra sẽ được canh lề ở phía bên phải.

Ví dụ:

WRITELN(x:5); WRITE(sin(13\*x):5:2);

|  |  |
| --- | --- |
| **Câu lệnh** | **Kết quả trên màn hình** |
| Writeln('Hello');  Writeln('Hello':10);  Writeln(500);  Writeln(500:5);  Writeln(123.457)  Writeln(123.45:8:2) | Hello  Hello  500  500  1.2345700000E+02  123.46 |

***6.3.2. Nhập dữ liệu***

Để nhập dữ liệu từ bàn phím vào các biến có kiểu dữ liệu chuẩn (trừ các biến kiểu BOOLEAN), ta sử dụng cú pháp sau đây:

**READLN(<biến 1> [,<biến 2>,...,<biến n>]);**

***Chú ý***: *Khi gặp câu lệnh* **READLN;***(không có tham số), chương trình sẽ dừng lại chờ người sử dụng nhấn phím ENTER mới chạy tiếp.*

**6.4. Các hàm và thủ tục thường dùng trong nhập xuất dữ liệu**

* Hàm **KEYPRESSED**: Hàm trả về giá trị TRUE nếu như có một phím bất kỳ được nhấn, nếu không hàm cho giá trị là FALSE.
* Hàm **READKEY**: Hàm có chức năng đọc một ký tự từ bộ đệm bàn phím.
* Thủ tục **GOTOXY(X,Y:Integer)**: Di chuyển con trỏ đến cột X dòng Y.
* Thủ tục **CLRSCR**: Xoá màn hình và đưa con trỏ về góc trên bên trái màn hình.
* Thủ tục **CLREOL**: Xóa các ký tự từ vị trí con trỏ đến hết dòng.
* Thủ tục **DELLINE**: Xoá dòng tại vị trí con trỏ và dồn các dòng ở phía dưới lên.
* Thủ tục **TEXTCOLOR(color:Byte)**: Thiết lập màu cho các ký tự. Trong đó color ∈ [0,15].

Thủ tục **TEXTBACKGROUND(color:Byte)**: Thiết lập màu nền cho màn hình.

**6.5 Cấu trúc chung của chương trình**

{Phần khai báo}

**Program** Ten\_chuong\_trinh ;

**Uses crt (\* khai báo thư viện \*)**

**Const (\* Khai báo hằng \*)**

**Var ….**

{Phần thân chương trình}

**Begin**

Câu lệnh 1;

Câu lệnh 2;

*...*

Readln;{ dừng màn hình, đọc kết quả}

**End.** (\* Kết thúc chương trình \*)

Ví dụ:

program CT\_3;

Begin

writeln (‘Chao cac ban’);

readln;

end.

**6.6 Các phím thông dụng**

|  |  |
| --- | --- |
| **Ctrl - Insert** | Đưa khối văn bản đang chọn vào Clipboard |
| **Shift-Insert** | Dán khối văn từ Clipboard xuống vị trí con trỏ |
| **F9** | Dịch chương trình |
| **Ctrl + F9** | Chạy chương trình |
| **Insert** | Chuyển qua lại giữa chế độ đè và chế độ chèn |
| **Home** | Đưa con trỏ về đầu dòng |
| **End** | Đưa con trỏ về cuối dòng |
| **Page Up** | Đưa con trỏ lên một trang màn hình |
| **Page Down** | Đưa con trỏ xuống một trang màn hình |
| **Ctrl-Home** | Đưa con trỏ về đầu văn bản |
| **Ctrl-End** | Đưa con trỏ về cuối văn bản |
| **Del** | Xoá ký tự ngay tại vị trí con trỏ |
| **Back Space** | Xóa ký tự bên trái con trỏ |
| **Ctrl-Y** | Xóa dòng tại vị trí con trỏ |

**BAØI TAÄP:**

+BT01: In ra màn hình theo dạng sau:

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

\* Họ tên: ……………………… \*

\* Lớp: …… \*

\* Môn thể thao ưa thích: \*

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

+BT02: Hãy viết biểu thức Pascal

|  |  |
| --- | --- |
| **Biểu thức trong tốn học** | **Hy viết biểu thức trong Pascal?** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

+BT003: Nhập vào năm sinh của 1 học sinh. Cho biết học sinh đó bao nhiêu tuổi.

+BT004: Nhập vào 1 số nguyên, in ra màn hình số lớn hơn số đó 10 đơn vị.

+BT005: Nhập vào 2 số A và B, in ra màn hình tổng của 2 số đó.

+BT006: Nhập vào 2 cạnh của 1 hình chữ nhật, cho biết diện tích của hình chữ nhật.

+BT007: Nhập vào 1 số, cho biết số liền trước của số đó.

+BT008: Nhập vào 2 số, in ra màn hình kết quả phép tính nhân của 2 số đó theo dạng:

60

x 2

------

120

+BT009: Nhập vào tổng số giây, đổi sang giờ, phút, giây**.**

* **Input:** tg (tổng số giây)
* **Output:** h (giờ), m (phút), s (giây)

+BT010: Nhập vào số tiền, cho biết số lượng vé số nhận được, biết cứ 10.000đ mua hàng được tặng 1 vé số.

+BT011: Nhập vào điểm Toán và điểm Văn, cho biết điểm trung bình của học sinh đó. Điểm trung bình chỉ lấy 2 số lẻ thập phân.

+BT012: Nhập vào 1 số, cho biết bình phương của số đó là bao nhiêu?

+BT013: Nhập vào giá trị X rồi tính giá trị của biểu thức sau: X2+2X+2

+BT014:Viết chương trình nhập 2 số nguyên sau đó đổi chỗ 2 số đó.

***Hướng dẫn:***

- Dùng các biến a, b để lưu hai số được nhập từ bàn phím;

- Gán cho biến tam giá trị của a.

- Gán giá trị của b cho a. (Sau lệnh này a có giá trị của b).

- Gán giá trị của tạm cho cho b (Sau lệnh này b có giá trị của tam = a).

+BT015:Nhập 2 số nguyên mỗi số có 3 ký số. Hãy in kết quả phép tính nhân

+BT016:Viết chương trình tính chu vi và diện tích của hình chữ nhật có chiều dài hai cạnh là a, b (được nhập từ bàn phím).

***Hướng dẫn:***

- Nhập hai cạnh vào hai biến a, b.

- Chu vi hình chữ nhật bằng 2\*(a+b); Diện tích hình chữ nhật bằng a\*b.

Tương tự:

a. Viết chương trình tính chu vi, diện tích hình vuông có cạnh a (được nhập từ bàn phím).

***Hướng dẫn:***

- Nhập cạnh vào biến canh.

- Chu vi hình vuông bằng 4\*canh; Diện tích hình vuông bằng canh\*canh.

b. Viết chương trình tính chu vi và diện tích hình tròn có bán kính r (được nhập từ bàn phím).

***Hướng dẫn:***

- Nhập bán kính vào biến r.

- Khai báo hằng : **Const** pi = 3.14

- Chu vi đường tròn bằng 2\*pi\*r.

- Diện tích hình tròn bằng π\*pi\*r.

+BT017:Viết chương trình nhập vào số nguyên N sau đó xuất ra màn hình các chữ số biểu diễn số đó.

Ví dụ: N=123 thì xuất ra màn hình là 00123.

+BT18: Viết chương trình tính chu vi, diện tích của tam giác có ba cạnh là a,b,c (được nhập từ bàn phím)

***Hướng dẫn:***

- Nhập ba cạnh của tam giác vào ba biến a,b,c.

- chu vi của tam giác p = a+b+c

- Diện tích của tam giác: s =.

+BT19: Viết chương trình cho phép tính trung bình cộng của bốn số.

***Hướng dẫn:***

- Nhập bốn số vào bốn biến a, b, c, d

- Trung bình cộng của a, b, c, d bằng (a + b + c + d)/4.

+BT20: Viết chương trình cho phép tính trung bình nhân của bốn số với điều kiện chỉ được sử dụng hai biến.

***Hướng dẫn:***

- Dùng một biến S có giá trị ban đầu bằng 1.

- Dùng một biến để nhập số.

- Sau khi nhập một số nhân ngay vào biến S.

- Trung bình nhân bốn số là căn bậc 4 tích của chúng (Dùng hai lần căn bậc hai).

+BT21: Viết chương trình cho biết chữ số hàng trăm, hàng chục, hàng đơn vị của một số có ba chữ số. Ví dụ khi nhập số 357 thì máy in ra:

- Chữ số hàng trăm: 3.

- Chữ số hàng chục: 5.

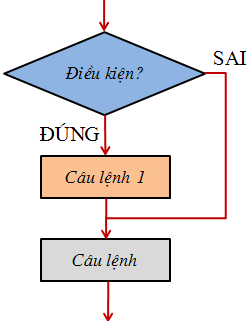
- Chữ số hàng đơn vị: 7.

**CHƯƠNG II: CÂU LỆNH CÓ CẤU TRÚC TRONG PASCAL**

I. CÂU LỆNH RẼ NHÁNH

* + - 1. **Dạng Thiếu**

|  |  |
| --- | --- |
| * + - * + Cú pháp: | **IF**<Điều kiện>**Then**<Câu Lệnh>; |



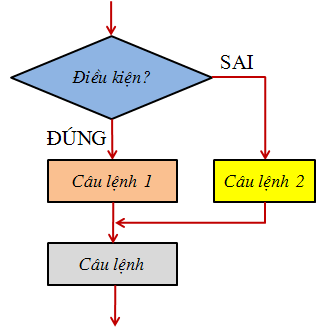
Ví dụ: nhập vào số tự nhiên a, in ra màn hình kết quả kiểm tra a có bé hơn 100 không?

if a<100 then write(‘dung’);

* + - 1. **Dạng Đủ**

*Không có dấu chấm phẩy*

|  |  |
| --- | --- |
| * + - * + Cú pháp: | **IF**<Điều kiện>**Then**<Câu Lệnh 1>  **Else**<Câu lệnh 2>; |



***Chú ý*:***Khi sử dụng câu lệnh IF thì đứng trước từ khoá ELSE không được có dấu chấm*

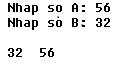
Ví dụ: nhập vào số tự nhiên a, in ra màn hình kết quả kiểm tra a<100 là đúng hay sai

if a<100 then write(‘dung’)

Else write(‘sai’);

|  |  |  |
| --- | --- | --- |
| ***Bài toán:***Nhập hai số nguyên *a, b* từ bàn phím. Yêu cầu in 2 số đó ra màn hình theo thứ tự tăng dần (không giảm). | | |
| ***Mô tả thuật toán:***  *Bước 1.* Nhập hai số nguyên *a, b* từ bàn phím;  *Bước 2.* Nếu *a* ≤ *b* thì hiển thị ra màn hình giá trị biến *a* trước rồi đến giá trị biến *b;*  *Bước 3.* Nếu *b < a*thì hiển thị ra màn hình giá trị biến b trước rồi đến giá trị biến *a;*  *Bước 4.Kết* thúc. | | |
| ***Viết chương trình:*** | |
| **Cách 1\_Điều kiện dạng thiếu** | **Cách 2\_Điều kiện dạng đủ** |
| **program** Sap\_xep1;  **var** A, B: integer;  **begin**  *{Buoc 1: Nhap hai so nguyen a, b tu ban phim}*  write('Nhap so A:'); readln(A);  write('Nhap so B:'); readln(B);  *{Buoc 2: Neu a < b thi hien thi ra man hinh gia tri bien a truoc roi den gia tri bien b}*  **if** A<=B **then** write(A,' ',B);  *{Neu b < a thi hien thi ra man hinh gia tri bien b truoc roi den gia tri bien a}*  **if** B<A **then** write(B,' ',A);  readln;  **End**. | **program** Sap\_xep2;  **var** A, B: integer;  **begin**  *{Buoc 1: Nhap hai so nguyen a, b tu ban phim}*  write('Nhap so A:'); readln(A);  write('Nhap so B:'); readln(B);  writeln;  *{Buoc 2: Neu a < b thi hien thi ra man hinh gia tri bien a truoc roi den gia tri bien b}*  **if** A<=B **then** write(A,' ',B)  **else** write(B,' ',A);  readln;  **End**. |

***Kết quả chạy chương trình:***



***Trắc nghiệm***

1. **Các** **câu lệnh Pascal sau đây được viết đúng hay sai?**

|  |  |  |
| --- | --- | --- |
|  | **Câu lệnh** | **True/False** |
|  | if x:=7 then a=b; |  |
|  | if x>5 then a:=b; m:=n; |  |
|  | if n>0 then begin a:=0; m:= –1 end else c:=a; |  |
|  | if x>5; then a:=b; |  |
|  | if x>5 then; a:=b; |  |
|  | if x>5 then a:=b; else m:=n; |  |
|  | If a>b then max : = a ; else max : = b ; |  |
|  | If a>b then max = a else max = b ; |  |
|  | If a>b then max : = a else max : = b ; |  |
|  | If a>b then max : = a esle max : = b ; |  |

1. **Sau mỗi câu lệnh sau đây, giá trị của biến *X* sẽ là bao nhiêu, nếu trước đó giá trị của *X* bằng 5?**

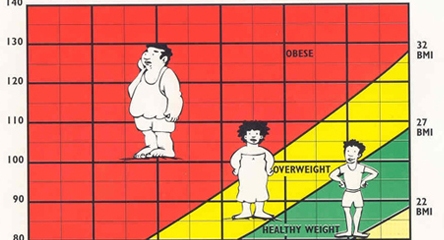
|  |  |
| --- | --- |
| **Câu lệnh** | **Giá trị của biến *X*** *(chọn đáp án đúng)* |
| 1. if (45 mod 3)=0 then X:=X+1; | a) X = 5 b) X=6 |
| 1. if X>10 then X:=X+1; | a) X = 5 b) X=6 |

**3. Viết các biểu thức điều kiện:**

|  |  |  |
| --- | --- | --- |
|  | a lớn hơn b |  |
|  | a bé hơn hoặc bằng b |  |
|  | a khác b |  |
|  | a chia hết cho b |  |
|  | a không chia hết cho b |  |
|  | a>=5 và a<=9 |  |
|  | a=5 hoặc a =9 |  |

**Bài tập**

+BT01: nhập vào chiều cao, cân nặng, kiểm tra cơ thể mập, vừa, ốm với công thức sau:

****BMI=cân nặng/(chiều cao)2

***Hướng dẫn:***

* Nếu BMI<18.5: coi chừng gió bay
* 18.5<Nếu BMI<25: người mẫu chuẩn

Nếu BMI>=25: Hơi béo một tí

+BT2 :Viết chương trình nhập vào một số nguyên và kiểm tra xem số vừa nhập là số chẵn hay số lẻ.

+BT3: Viết chương trình in ra số lớn hơn trong hai số (được nhập từ bàn phím).

***Hướng dẫn:***

* Nhập hai số vào hai biến a, b.
* Nếu a > b thì in a. Nếu a <= b thì in b.
* Hoặc: Nếu a > b thì in a. Ngược lại thì in b.

+BT4 Nhaäp vaøo 2 soá, cho bieát caên baäc hai cuûa hieäu 2 soá ñoù. Neáu khoâng tính ñöôïc caên baäc 2 thì thoâng baùo cho ngöôøi söû duïng bieát laø “khoâng tính ñöôïc.

+BT5: Viết chương trình in ra số lớn nhất(hoặc nhỏ nhất) trong bốn số(hoặc n số) nhập từ bàn phím.

+BT6:Viết chương trình xét xem một tam giác đó là tam giác gì

+BT7: Viết chương trình xét xem một tam giác có là tam giác vuông hay không khi biết ba cạnh của tam giác.

+BT8: Viết chương trình giải phương trình ax + b = 0 (Các hệ số a, b được nhập từ bàn phím).

***Hướng dẫn:***

* Nếu a ≠ 0 thì phương trình có nghiệm x = 
* Nếu a = 0 và b = 0 thì phương trình có vô số nghiệm
* Nếu a = 0 và b ≠ 0 thì phương trình vô nghiệm

+BT9: Viếtchươngtrìnhgiải phươngtrìnhbậc2(ax2 + bx +c =0)

***Hướng dẫn:***

TH1: a=0,b=0,c=0: Phương trình có vô số nghiệm

TH2: a=0,b=0,c≠0:Phương trình vô nghiệm

TH3: a=0,b≠0,c≠0: Phương trình có nghiệm x= -c / b

TH4: a≠0,b≠0,c≠0:

Tính delta=b2 – 4ac

delta < 0:Phương trình vô nghiệm

delta = 0: Phương trình có nghiệm kép x= -b / 2a

delta > 0: Phương trình có 2 nghiệm x= (-b + ) / 2a và x= (-b - ) / 2a

**II. CU LỆNH LỰA CHỌN CASE .. OF**

***Dạng 1:***

+ Cú pháp:

**CASE** biến **OF**

Giá trị 1 : lệnh 1 ;

Giá trị 2 : lệnh 2 ;

………………………

**END;**

+ Ý nghĩa: Nếu giá trị của biến = giá trị 1 thì thực hiện lệnh 1, nếu giá trị của biến = giá trị 2 thì thực hiện lệnh 2, …

***Dạng 2:***

+ C php:

**CASE** biến **OF**

Giá trị 1 : lệnh 1 ;

Giá trị 2 : lệnh 2 ;

………………………………

**ELSE**

Lệnh n+1;

**END;**

+ Ý nghĩa: Nếu giá trị của biến = giá trị 1 thì thực hiện lệnh 1, nếu giá trị của biến = giá trị 2 thì thực hiện lệnh 2, …Nếu giá trị của biến không = giá trị nào hết thì thực hiện lệnh n.

**Bài tập**

+BT1: Viết chương trình nhập vào một tháng nào đó , máy sẽ thông báo tháng đó có bao nhiêu ngày.

***Hướng dẫn:*** Tháng 2:28 ngày

Tháng 1,3,5,7,8,10,12: 31 ngay

Tháng 4,6,9,11: 30 ngay

+BT2: Nhập vào điểm trung bình của 1 học sinh cho biết học sinh xếp loại gì

***Hướng dẫn:***dtb 9,10: Gioi

dtb7,8: Kha

dtb5,6:Tbinh

còn lại: Yeu

+BT3: Viết chương trình dịch các ngày trong tuần sang tiếng anh

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday |

***Hướng dẫn:***

* Dùng biến a kiểu byte để chứa thứ (2 đến 8)
* Trường hợp a = 2: Monday
* Trường hợp a = 3: Thursday
* ...
* Trường hợp a = 7: Sunday

+BT4: Viết chương trình nhập vào tuổi của một người và cho biết người đó là thiếu niên, thanh niên, trung niên hay lão niên. Biết rằng: *nếu tuổi nhỏ hơn 18 là thiếu niên, từ 18 đến 39 là thanh niên, từ 40 đến 59 là trung niên và lớn hơn 60 là lão niên..*

+BT5: Nhập vào năm cho biết năm âm lịch

***Hướng dẫn:***Xuất ra màn hình **Can Chi**

N mod 10 = **can**:

+ 0: Canh

+ 1: Tân

+ 2: Nhâm

+ 3: Quý

+ 4: Giáp

+ 5: Ất

+ 6: Bính

+ 7: Đinh

+ 8: Mậu

+ 9: Kỷ

N mod 12 = **chi:**

+ 0: Thân

+ 1: Dậu

+ 2: Tuất

+ 3: Hợi

+ 4: Tý

+ 5: Sửu

+ 6: Dần

+ 7: Mẹo

+ 8: Thìn

+ 9: Tỵ

+ 10: Ngọ

+ 11: Mùi

+BT6: Nhập vào số tiền (3 chữ số) in ra số tiền bằng chữ

***Hướng dẫn*:** 235 ghi hai trăm ba mươi lăm

**iii. Lệnh vòng lặp For .. DO**

***Dạng 1:***

**FOR** biến **: =** giá trị đầu **TO** giá trị cuối **DO**

**Begin**

Lệnh 1 ;

Lệnh 2 ;

………

Lệnh n ;

**End;**

***Dạng 2***

**FOR** biến **: =** giá trị cuối **DOWNTO** giá trị đầu **DO  
 Begin**

Lệnh 1 ;

Lệnh 2 ;

………

Lệnh n ;

**End;**

**Bài tập**

+ BT1: In ra màn hình các số nguyên từ 1 🡪 15, từ 15🡪1

+ BT2: In ra màn hình 5 ngôi sao \* theo hàng ngang và hàng dọc

+ BT3:Nhập 1số nguyên N. In ra màn hình các sốchẵn( hoặc số lẻ) từ 1🡪N

+ BT4:Nhập 1số nguyên N. In ra màn hình Giai thừa của số đó.

VD: 3!=1\*2\*3=6

+ BT5:Nhập 1số nguyên N. In ra màn hình tổng các số từ 1🡪N

VD: N=3 kết quà:1+2+3=6

+ BT6:Nhập vào 2 số a,b. In ra màn hình tổng các số từ a🡪b

VD: a=3, b=5 kết quả 3+4+5=12

+ BT7:Nhập 1số nguyên dương N. In ra màn hình các ước số của nó

+ BT8:Nhập 1số nguyên N. In ra màn hình các số nguyên tố từ 1🡪N

+ BT9: In ra màn hình bảng cửu chương số N bất kỳ. Với N nhập từ bàn phím. Có dạng sau:

2x1 = 2

2x2 = 4

……...

2x10=20

+ BT10: In ra màn hình dạng sau:

1234

1234

1234

1234

+ BT11: In ra màn hình dạng sau:

1 1 1 1

2 2 2 2

3 3 3 3

4 4 4 4

+ BT12: In ra màn hình dạng sau:

1234

2 3 4 1

3 4 1 2

4 1 2 3

+ BT13: In ra màn hình dạng sau:

1234

2 3 4 5

3 4 5 6

4 5 6 3

+ BT14: In ra màn hình dạng sau:

12 …………9

10 11 …...…..19

……………..…

90 91……......99

+ BT15: In ra màn hình dạng sau:

\* \* \* \*

\* \* \* \*

\* \* \* \*  
\* \* \* \*

+ BT16: In ra màn hình dạng sau:

\*

\* \*

\* \* \*  
\* \* \* \*

+ BT17: In ra màn hình dạng sau:

\*

\* \*

\* \* \*  
\* \* \* \*

+ BT18: In ra màn hình dạng sau:

\* \* \* \*

\* \* \*

\* \*  
 \*

+ BT19: In ra màn hình dạng sau:

\*

\* \* \*  
 \* \* \* \* \*

+ BT20: In ra màn hình dạng sau:

\* \* \* \*

\* \*

\* \*  
\* \* \* \*

**II. Lệnh REPEAT … UNTIL**:

* + Là vòng lặp không xác định trước được số lần lặp (số lần lặp tối thiểu là 1).
  + Cấu trúc:

**REPEAT**

BEGIN

Các câu lệnh cần sử dụng;

END;

**UNTIL** điều kiện;

* + Ý nghĩa: lặp lại các câu lệnh cho đến khi thỏa mãn điều kiện.

**III. Lệnh WHILE … DO**:

* + Là vòng lặp không xác định trước được số lần lặp (số lần lặp tối thiểu là 0).
  + Cấu trúc:

**WHILE** điều kiện **DO**

BEGIN

Các câu lệnh cần sử dụng;

END;

* + Ý nghĩa: Trong khi điều kiện còn đúng thì thực hiện vòng lặp.

***Lưu ý: i:=i+1 có thể viết INC(i)***

***i:= i-1 có thể viết DEC(i)***

**Bài tập**

+ BT1: In ra màn hình các số nguyên từ 1 🡪50

+ BT2:Nhập 1số nguyên N. In ra màn hình các số chẵn( hoặc số lẻ) từ 1🡪N

+ BT3:Nhập 1số nguyên N. In ra màn hình Giai thừa của số đó.

VD: 3!=1\*2\*3=6

+ BT4:Nhập 1số nguyên N. In ra màn hình tổng các số từ 1🡪N. Cho phép chạy lại từ đầu khi đưa ra câ hỏi”ban co muon tiep tuc nua khong ?”. Nếu chọn “c” chương trình tiếp tục chạy với kết quả khác. Chọn “K” sẽ thoát khỏi chương trình

VD: N=3 kết quà:1+2+3=6

+ BT5:Nhập 1số nguyên dương N. In ra màn hình các ước số của nó.

+ BT6:Nhập 1số nguyên N. In ra màn hình các số chính phương từ 1🡪N

***Hướng dẫn:***

- Cho biến i chạy từ 1 đến a div 2.

- Nếu i2=N thì a là số chính phương

- Nếu i2= a div 2 thì a không là số chính phương

+ BT7Nhập 1số nguyên N. In ra màn hình các số chia hết cho 3 từ 1🡪N

+ BT8:Nhập 1số nguyên N. In ra màn hình các số chia hết cho 4 tận cùng bằng 6 từ 1🡪N

+BT9: Nhập 1số nguyên N.Viết chương trình in ra các số lẻ nhỏ hơn hoặc bằng N (Với N được nhập).

***Hướng dẫn:***

- Cho biến i chạy từ 1 đến N.

- Nếu i chẵn ( i chia 2 dư 0) thì in ra số n.

+BT10: Nhập 1số nguyên N.Viết chương trình in ra các số lẻ nhỏ hơn hoặc bằng N ( Với N được nhập). Sao cho 15 số lẻ được in trên một dòng.

***Hướng dẫn:***

- Cho j =0.

- Cho biến i chạy từ 1 đến N.

- Nếu i chẵn ( i chia 2 dư 0) thì in ra số n và tăng dem lên 1

- Nếu dem chia hết cho 15 thì thực hiện xuống dòng (Dùng Writeln).

+ BT11: Viết chương trình xét xem một số N được nhập từ bàn phím có phải là số hoàn chỉnh không.

***Hướng dẫn:***

- Dùng biến n lưu số cần xét.

- Biến S có giá trị ban đầu bằng 0.

- Cho i chạy từ 1 đến N-1. nếu i là ước của n thì cộng thêm i vào S.

- Nếu S = N thì S là số hoàn chỉnh.

+BT12: Viết chương trình xét xem một số n có phải là số nguyên tố không?

***Hướng dẫn:***

- Nếu n không chia hết mọi số i có giá trị từ 2 đến n - 1 thì n là số nguyên tố.

- Sử dụng biến ok có kiểu boolean và có giá trị ban đầu là true.

- Cho biến i chạy từ 2 đến N – 1. Xét n mod i. Nếu bằng 0 thì gán ok = false. Ngược lại vẫn để nguyên ok.

+BT13:Nhập 1số nguyên N. In ra màn hình các số nguyên tố từ 1🡪N

+BT14:Viết chương trình in ra các số lẻ nhỏ hơn hoặc bằng số nguyên dương N( Với N được nhập). Yêu cầu nhập lại nếu N<=0

***Hướng dẫn:***

- Sử dụng kiến thức số lẻ đầu tiên bằng 1. Số lẻ sau bằng số trước cộng với 2.

- Cho biến i có giá trị ban đầu bằng 1.

- Dùng vòng lặp while do với điều kiện i <N và công việc bên trong là in i và tăng i lên 2.

+BT15: Nhập vào 1 số thực a, tìm số nguyên N nhỏ nhất thỏa:

1+1/2+1/3+….+1/N > a

**IV. MẢNG**

**1. Khai báo mảng 1 chiều:**

Var tên mảng **: array [**giới hạn dưới **..** giới hạn trên**] of** kiểu dữ liệu;

*\* Khai báo 1 kiểu mảng:*

**Type** kiểu mảng **= array [**giới hạn dưới **..** giới hạn trên**] of** kiểu dữ liệu;

**Var** tên mảng: tên kiểu mảng;

VD: var A:array[1..20] of integer;

**🡪**Mảng số nguyên gồm 20 phần tử

**BÀI TẬP**

+BT01: Viết chương trình nhập và in ra mảng các số nguyên gồm N phần tử.

+BT02: Viết chương trình in ra phần tử vị trí thứ 3 trong mảng gồm N phần tử.

+BT03: Viết chương trình in ra mảng các số lẻ (chẵn) gồm N phần tử.

+BT04: Viết chương trình in ra mảngcác số nguyên chia hết cho 5 gồm N phần tử.

+BT05: Viết chương trình in ra mảnggồm tổng các số nguyêngồm N phần tử.

+BT06: Viết chương trình in ra trung bình cộng của mảnggồm tổng các số nguyêngồm N phần tử.

+BT07: Viết chương trình in ra mảnggồm tổng các số nguyên chẵn(lẻ) gồm N phần tử.

+BT08: Viết chương trình in ra mảnggồm tích các số nguyên chẵn(lẻ) gồm N phần tử.

+BT09: Viết chương trình in ra mảnggồm N phần tử được sắp xếp tăng dần(giảm dần).

+BT10: Viết chương trình in ra mảngcác số nguyên tố gồm N phần tử

+BT11: Viết chương trình in ra phần tử có giá trị lớn nhất(nhỏ nhất) trong mảng.

+BT12: Nhập vào 1 số nguyên x.Viết chương trình đếm xem x xuất hiện trong mảnggồm N phần tử mấy lần.

**2. Khai báo mảng 2 chiều:**

Var tên mảng **: array [1.. số dòng,1..số cột] of** kiểu dữ liệu;

VD: Mảng 2 chiều với n=4, m=4

7 6 4 2

5 4 9 8

2 1 1 0

5 4 2 3

**BÀI TẬP**

1. Nhập mảng 2 chiều gồm n dòng, m cột (n x m). In mảng theo dạng ma trận.
2. In ra dòng 1 của ma trận
3. Tính tổng các số nguyên trên mỗi dòng(cột).
4. Tính tích các số nguyên trên mỗi cột(dòng)..
5. Tìm số lớn nhất trên mỗi cột(dòng).
6. Tìm số lớn nhật trong mảng

**CHƯƠNG III: HÀM VÀ THỦ TỤC TRONG PASCAL**

1. **THỦ TỤC**:

Khai báo 1 thủ tục: có cấu trúc như là 1 chương trình chính

**PROCEDURE**tên ;

BEGIN

Các lệnh dùng trong thủ tục;

END;

Ví dụ:

PROCEDURE tong ;;

BEGIN

kq:= x+y ;

END;

**II. HÀM**:

Khai báo 1 hàm: có cấu trúc như là 1 chương trình chính. Hàm luôn trả về 1 giá trị

**FUNCTION** tên : kiểu dữ liệu trả về;

BEGIN

Các lệnh dùng trong hàm;

…………………………

Tên hàm := giá trị ;

END;

Ví dụ:

FUNCTION tong : integer;

BEGIN

tong:= x+y ;

END;

**CẤU TRÚC CHUNG**

PROGRAM Tên\_chương\_trình;

USES CRT;

CONST ............;

TYPE ............;

VAR ............;

PROCEDURE THUTUC;

[Khai báo Const, Type, Var]

BEGIN

..............

END;

FUNCTION HAM;

[Khai báo Const, Type, Var]

BEGIN

..............

HAM:=<Giá trị>;

END;

BEGIN

{Chương trình chính}

...................

THUTUC;

...................

A:= HAM;

...................

END.

**BÀI TẬP**

+BT01: Viết 1 thủ tục nhập mảng sốnguyên, thủ tục xuất mảng sô nguyên gồm 100 phần tử

+BT02: Viết 1 thủ tục nhập mảng số nguyên gồm 100 phần tử. In ra màn hình 1 mảng được sắp xếp theo thứ tự tăng dần.

+BT03: Nhập 1 số nguyên X bất kỳ. Cài đặt 1 hàm kiểm tra X có phải số chẵn hay không ?

+BT04: Nhập 1 số nguyên X bất kỳ. Cài đặt 1 hàm kiểm tra X có phải số nguyên tố hay không

+BT05: Viết 1 thủ tục nhập mảng số nguyên, thủ tục xuất mảng tổng các số nguyên tố trong mảng

+BT06: Viết 1 thủ tục nhập mảng số nguyên gồm 100 phần tử. In ra màn hình tổng mảng vừa nhập

+BT07: Viết 1 thủ tục nhập mảng số nguyên gồm 100 phần tử. In ra màn hình tổng các số nguyên dương chia hết cho 5

+BT08: Viết 1 thủ tục nhập mảng số nguyên gồm 100 phần tử. Đếm số phần tửA[i] trong mang bằng giá trị X

+BT09: Viết 1 thủ tục nhập mảng số nguyên gồm 100 phần tử. Đếm số lần phần tử X xuất hiện trong mảng

+BT10: Viết 1 thủ tục nhập mảng số nguyên gồm 100 phần tử. Tìm kiếm phần tử X trong mảng và thay thế phần tử Y

+BT11: Viết chương trình in ra thực đơn sau

1. Nhập mảng gồm n số
2. Sắp xếp mảng tăng dần
3. Thêm 1 số m vào cuối(hoặc đầu) mảng

Hướng dẫn:

*+ Nhập số cần thêm x*

*+ Begin*

*A[n+1]:=x;*

*n:=n+1;*

*end;*

1. Xóa 1 số tại vị trí k trong mảng

Hướng dẫn:

*+ Nhập vị trí k cần xóa*

*+ for i:=k to n-1 do*

*Begin*

*A[i]:=A[i+1];*

*n:=n-1;*

*end;*

1. Sửa 1 số tại vị trí k trong mảng

Hướng dẫn:

*+ Nhập vị trí k trong mảng*

*+ Nhập số mới x*

*+ A[k]:=x*

1. Tích các số nằm trong khoảng [a,b]

Hướng dẫn

*+ khai báo biến m*

*+ Nhập cận dưới a, cận trên b*

*+ for i:=1 to n do*

*If(A[i]>=a and A[i]<=b then*

*Begin {thêm A[i] vào b}*

*m:=m+1;*

*b[m]:=A[i];*

*end;*

*+ for i:=1 to m do {xuất mảng b[i] chứa các số trong khoảng [a,b]*

*Writeb[i];*

**Chương IV: KIỂU KÝ TỰ – KIỂU CHUỖI**

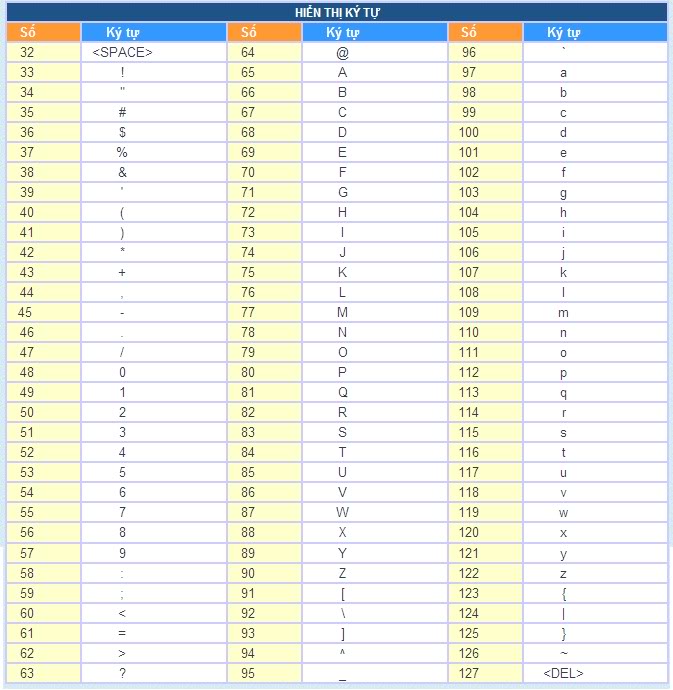
1. **KIỂU KÝ TỰ**:

Kiểu char dùng để biểu diễn các ký tự thuộc bảng chữ cái, chữ số và ký tự đặc biệt.  
Để biểu diễn thông tin ta cần sắp xếp các ký tự theo một bảng, thông thường ta sắp xếp theo bảng mã ASCII. Bảng mã ASCII có 256 ký tự, mỗi ký tự được gán mã số từ 0 đến 255.

**Cú pháp**: Var tên biến: **Char**

*Ví dụ: Var c1, c2 : char;*

***Hàm sử dụng :***

* + Hàm **ORD**(ký tự) : cho biết mã ASCII của ký tự.
  + Hàm **CHR**(n) : cho biết ký tự có mã ASCII là n là ký tự nào.
  + **Hàm **UPCASE**(ký tự) : chuyển ký tự chữ thường sang chữ hoa.

1. **KIỂU CHUỖI**:

String là kiểu dữ liệu có cấu trúc dùng để xử lý các chuỗi ký tự. Nó có nhiều điểm gần giống với mảng nhưng khác là số phần tử của mảng thì cố dịnh còn của xâu ký tự thì có thể thay đổi.

***Cú pháp:***

**TYPE** TênKiểu = **STRING[độ dài của xâu];**

**VAR**  Tên biến : TênKiểu;

hoặc khai báo biến trực tiếp:

VAR Tên biến : **STRING[độ dài của xâu]**;

**Ví dụ:**

*Type      Hoten  = String[30];*

*St80 = String[80];*

*Var        Name : Hoten;*

*Line : St80;*

*St : String; {St có tối đa là 255 ký tự}*

**1. Truy xuất dữ liệu kiểu string**

     - Có thể sử dụng các thủ tục xuất nhập Write, Writeln, Readln để truy xuất các biến kiểu String.

*Ví dụ:   
Readln(st);  
Writeln(st);*

     - Để truy xuất đến ký tự thứ k của xâu ký tự, ta sử dụng cú pháp sau: Tênbiến[k].

**2. Các phép toán trên xâu ký tự**

a. Phép nối xâu: +

*Ví dụ:*

*st1:=’Le’; st2:=’Thanh’; St=st1 + st2;*

*-> KQ: ‘Le Thanh’*

b. Các phép toán quan hệ: =, <>, <, <=, >, >=.

3. Các thủ tục và hàm vế xâu ký tự

a. Hàm **length(st):** cho độ dài thực của xâu ký tự

*ví dụ: st:=’le thanh’ thì LENGTH(st) cho bằng 8.*

b/ Thủ tục **DELETE(st, pos, num):** xóa num ký tự trong xâu st kể từ vị trí pos

*Ví dụ: st= ‘FILENAME’*

*Delete(st,5,4) lúc đó st cho ra là ‘FILE’*

c/ Thủ tục **INSERT(obj, st, pos)**: Thủ tục cho kết quả bằng cách chèn xâu ký tự có tên là Obj vào xâu st tại vị trí pos, những ký tự đứng sau pos sẽ được dời vềphía sau của xâu ký tự obj.

*Ví dụ: obj:= ‘Thanh ‘*

*st:=’Le Lam’;*

*INSERT(obj,st,4) lúc đó st=’Le Thanh Lam’;*

d/ Thủ tục **STR(value, st)**: Thủ tục này thực hiện việc chuyển đối giá trị kiểu số(value) sang dạng xâu ký tự và gán cho biến st.

*Ví dụ: n là một só nguyên có giá trị: n:=150;*

*STR(n:5,st) sẽ cho kết quả xâu st là: st=’ 150’;*

e/ Thủ tục **VAL(st, value,code)** đối một xâu ký tự st sang dạng số và gán cho biến value, nếu biến đối thành công thì code sẽ nhận giá trị bằng 0. ngược lại thì cho giá trị khác không

*Ví dụ: VAL(‘123’,value,code) lúc này code sẽ nhận giá trị bằng 0 và value=123*

f/ Hàm **COPY(st, pos, num)**: sao chép trong xâu st, num ký tự tại vị trí pos,

*Ví dụ: st=’Le Thanh Lam’*

*COPY(st,4,5) = ‘Thanh’;*

g/ Hàm **CONCAT(s1,s2,…,sn):** hàm cho ra 1 xâu mới bằng cách nối đuôi các xâu s1,s2,…,sn lại với nhau.

*Ví dụ: CONCAT(‘Le ’,’Thanh ‘, ‘Lam’) = ‘Le Thanh Lam’;*

h/ Hàm **POS(st1,st2)**: hàm cho tavị trí tìm thấy đầu tiên của xâu s1 trong xâu s2.

*Ví dụ: POS(‘Lam’,‘Le Thanh Lam’) = 10;*

i/ Hàm **UPCASE(st)**

*Ví dụ: UPCASE(Ký\_tự)--> Đổi Ký\_tự thành "KÝ\_TỰ" in hoa*

**BÀI TẬP**

+BT01: Viết chương trình nhập vào một xâu ký tự từ bàn phím. Đổi xâu ký tự đó sang chữ in hoa rồi in kết quả ra màn hình.

Ví dụ :Xâu abcdAbcD sẽ cho ra xâu ABCDABCD.

+BT02: Viết chương trình nhập vào một xâu ký tự từ bàn phím. Đổi xâu ký tự đó sang chữ thường rồi in kết quả ra màn hình.

+BT03: Viết chương trình đếm số ký tự chữ số trong một xâu ký tự được nhập vào từ bàn phím.

+BT04: Viết chương trình nhập vào một xâu ký tự từ bàn phím. Tìm xâu đảo ngược của xâu đó rồi in kết quả ra màn hình

+BT05: Viết chương trình nhập một xâu từ bàn phím. In ra xâu đó sau khi xóa hết các ký tự trắng thừa trong xâu.

+BT06: Viết chương trình liệt kê các từ của một xâu ký tự được nhập vào từ bàn phím

+BT07: Viết chương trình xóa các ký tự chữ số trong một xâu ký tự được nhập vào từ bàn phím.

+BT08: Nhập chuỗi gồm cả chữ và số. Xuất ra màn hình các số riêng và các chữ riêng.

**Chương V: Tệp trong pascal – Kiểu file**

**Khái niệm về tệp:**

Tệp là một dãy các phần tử cùng kiểu được sắp xếp một cách tuần tự. Tệp dữ liệu được lưu trữ ở bộ nhớ ngoài dưới một tên nào đó.

Tệp tập hợp trong nó một số phần tử dữ liệu có cùng cấu trúc giống như mảng nhưng khác mảng là số phần tử của tệp chưa được xác định.

Trong Pascal có 3 loại tệp được sử dụng là:

   1. Tệp có kiểu:

Tệp có kiểu là tệp mà các phần tử của nó có cùng độ dài và cùng kiểu dữ liệu.

   2 Tệp văn bản:

Dùng để lưu trữ dữ liệu dưới dạng các ký tự của bảng mã ASCII, các ký tự này được lưu thành từng dòng, độ dài các dòng có thể khác nhau. Ví dụ 2008 (kiểu word) khi ghi vào tệp văn bản cần 4 Byte (không phải 2 Byte).

   3 Tệp không kiểu:

Tệp không kiểu là một loại tệp không cần quan tâm đến kiểu dữ liệu ghi trên tệp. Dữ liệu ghi vào tệp không cần chuyển đổi.

 Tác dụng lớn nhất của kiểu dữ liệu tệp là ta có thể lưu trữ các dữ liệu nhập vào từ bàn phím và các kết quả xử lý trong bộ nhớ RAM ra tệp để dùng nhiều lần.

**1. Khai báo:**

Var f: **file of** integer;

f: file of hocsinh;

**2. Các lệnh kiểu tệp**

a. Đặt tên file

**Assign**(f,’Ho so.dat’);

b. Mở file ra đọc

**Reset**(f);

c. Mở file ra ghi

**Rewrite**(f); {khởi tạo tệp rỗng}

d. Đọc dữ liệu từ file f ra biến x

**Read**(f,x);

e. Ghi dữ liệu từ biến x vào file

**Write**(f,x);

f. Đổi tên file

**Rename**(f,’new.dat’);

g. Di chuyển con trỏ đến vị trí p

**Seek**(f,p);

h. Lấy số phần tử của file

n:=filesize

i. Đóng file

**Close**(f);

**Lưu ý:**

**. Pas**: là tệp chứa chương trình viết bằng Pascal .  
  
**\* . Doc**: là tệp chứa văn bản, tài liệu .  
  
**\* . Txt** : chứa tệp văn bản .  
  
**\* . Com và \* . Exe**: là các tệp chứa chương trình dưới dạng mã mà máy có thể chạy ngay được trên máy tính .  
  
**\* . Dat**: chứa các dữ liệu cần xử lý ( Data )

Ví dụ:

Program TAO\_TEP\_1 ;

Var

I : Integer ;

F : File Of Integer ;

BEGIN

Assign ( F, 'Nguyen.dat' ) ;

Rewrite ( F ) ;

For I :=1 To 100 Do Write ( F, I ) ;

Close ( F ) ;

END .

**BÀI TẬP**

+BT01:Hãy lập chương trình tạo một tệp số nguyên chứa các số nguyên tố nhỏ hơn 10000

theo thứ tự tăng dần

+BT02:Cho f là tệp văn bản chứa các xâu 10 kí tự.

Hãy lập chương trình nhập và hiển thị nội dung file đó lên màn hình, mỗi xâu một dòng,đầy trang màn hình thì dừng lại đợi gõ Enter mới hiển thị trang tiếp theo cho tới hết.

+BT03: Bạn hãy viết chương trình cho phép đọc dữ liệu từ bàn phím và ghi thêm vào cuối

một tệp các bản ghi

+BT04:Cho một văn bản chứa trong một text file f.Trong văn bản,tính từ trái sang phải,từ trên xuống dưới,kí tự # là kí hiệu xoá đi một từ đứng ngay trước nó nếu có.

Ví dụ

‘#Ta#oi di ngu#h###hoc’ có nghĩa là ‘Toi di hoc’

Bạn hãy viết chương trình sửa lại file f theo quy ước trên

+BT05: Cho 2 file f và g cùng kiểu ( nhưng không rõ kiểu nào ).

Bạn hãy lập thủ tục gán nội dung của file g cho file f.

+BT06: Cho một file text ghi lại một chương trình Pascal của một học sinh.

Hãy viết chương trình kiểm tra lỗi của chương trình Pascal trên theo các cách sau đây:

Cách 1:

Kiểm tra xem số lượng các dấu ‘ ( dấu mở và dấu đóng ) có bằng nhau không ?

Cách 2:

Kiểm tra xem số lượng các từ Begin và End có bằng nhau không ?

+BT07: Cho một file text.

Hãy viết chương trình đếm xem file text trên chứa bao nhiêu từ.( Chú ý:theo quy định,các từ cách nhau bởi một hay nhiều dấu cách )

+BT08: Cho một file text.

Viết chương trình loại bỏ các khoảng trống thừa bên trong file text này