

NGUYỄN TRỌNG MINH HỒNG PHƯỚC

# NGUYÊN NGỮ LẬP TRÌNH PASCAL

(Bổ trợ môn Tin học 11)

08/2014

“

Sự lười biếng của bản thân như một cái rễ cây.  
Chúng nhanh chóng phát triển  
và ghì chặt bạn tại một chỗ.

Hãy ghim chặt bản thân tại một chỗ.”

”

Chúng nhanh chóng phát triển

và ghì chặt bạn tại một chỗ.

# LỜI GIỚI THIỆU

*Pascal* là một ngôn ngữ lập trình cấp cao do *Niklaus Wirth*, giáo sư điện toán trường đại học Kỹ thuật Zurich (Thụy Sĩ) thiết kế và công bố vào năm 1971 và đặt tên là Pascal để tưởng niệm nhà Toán học và Triết học nổi tiếng Blaise Pascal (người Pháp).

Ban đầu, Pascal là một ngôn ngữ được hướng để dùng trong giảng dạy về lập trình có cấu trúc, và nhiều thế hệ sinh viên đã vào nghề thông qua việc học Pascal như ngôn ngữ vỡ lòng trong các chương trình học đại cương. Nhiều biến thể của Pascal ngày nay vẫn còn được sử dụng khá phổ biến, cả trong giảng dạy lẫn trong công nghiệp phát triển phần mềm.

Ngày nay, ngôn ngữ lập trình Pascal được sử dụng để giảng dạy trong các trường phổ thông, đặc biệt là ở môn Tin học lớp 8 và Tin học lớp 11. Ngoài ra, Pascal còn được sử dụng trong các ứng dụng thực tế trong cuộc sống như lập trình trong các ngành kỹ thuật, công nghiệp.

Với mong muốn giúp các bạn có được những kiến thức cơ bản nhất về ngôn ngữ lập trình cơ bản này, tạo tiềm đề học tập nghiên cứu về sau, nên tôi biên soạn ra quyển sách này. Được cấu trúc gồm các phần Lý thuyết, Bài tập vận dụng, Ôn tập, Mở rộng, mong rằng sẽ giúp đỡ cho các bạn, nhất là trong môn Tin học lớp 11.

Chân thành cảm ơn các thầy cô đã từng giảng dạy tôi, cho tôi những kiến thức cơ bản nhất, để tôi có đủ vốn kiến thức biên soạn ra quyển sách này. Tuy nhiên, với vốn kiến thức có hạn, khả năng lập trình hạn chế, nên trong quá trình biên soạn chắc chắn sẽ gặp sai sót, rất mong sự đóng góp từ các bạn.

THPT Chuyên Hùng Vương – Bình Dương  
Tháng 08 năm 2014  
Nguyễn Trọng Minh Hồng Phước





# CHƯƠNG 1. MỘT SỐ KHÁI NIỆM VỀ NGÔN NGỮ LẬP TRÌNH

## BÀI 1: KHÁI NIỆM

Chương trình dịch: chương trình đặc biệt có chức năng chuyển đổi chương trình được viết bằng ngôn ngữ lập trình bậc cao thành chương trình thực hiện được trên máy tính. Chương trình dịch có hai loại: *thông dịch* và *biên dịch*.

- Thông dịch (interpreter):

- + Kiểm tra tính đúng đắn của câu lệnh tiếp theo trong chương trình nguồn.
- + Chuyển đổi thành câu lệnh trong ngôn ngữ máy.
- + Thực hiện các câu lệnh vừa chuyển đổi.

- Biên dịch (compiler):

- + Duyệt, phát hiện lỗi, kiểm tra tính đúng đắn của các câu lệnh trong chương trình nguồn.
- + Dịch chương trình nguồn thành chương trình đích có thể thực hiện trên máy tính và có thể lưu trữ.

## BÀI 2: CÁC THÀNH PHẦN CƠ BẢN CỦA NGÔN NGỮ LẬP TRÌNH

### I/ Thành phần cơ bản:

- *Bảng chữ cái*: các chữ cái in thường và in hoa của bảng chữ cái tiếng Anh, 10 chữ số Ả-rập, các kí tự đặc biệt (+, -, \*, /, ...).
- *Cú pháp*: là bộ quy tắc để viết chương trình.
- *Ngữ nghĩa*: xác định ý nghĩa theo tác cần phải thực hiện, ứng với tổ hợp kí tự dựa vào ngữ cảnh của nó.

### II/ Một số khái niệm:

- *Tên*: không chứa dấu cách, bắt đầu bằng chữ số, chứa các kí tự không hợp lệ.
  - + Tên dành riêng: program, uses, const, type, var, ...
  - + Tên chuẩn: abs, integer, real, sqr, longint, ...
  - + Tên do người lập trình đặt: delta, VD, ...
- *Hằng*: là đại lượng có giá trị không đổi trong suốt quá trình thực hiện chương trình, gồm hằng số học (vd: 2, 0, -2.236E01, ...), hằng lô-gic (true hoặc false), hằng xâu (vd: 'information', 'lop 11L', ...).
- *Biến*: là đại lượng để lưu trữ giá trị và có thể thay đổi trong quá trình thực hiện chương trình.
- *Chú thích*: được đặt trong {...} hoặc (\*...\*).

## CHƯƠNG II. CHƯƠNG TRÌNH ĐƠN GIẢN

### BÀI 1: CẤU TRÚC CHƯƠNG TRÌNH

#### I/ Phần khai báo:

- Tên chương trình: **Program tên\_chương\_trình;**
- Thư viện: **Uses tenthuvien;**

Chú ý: đối với thư viện CRT, muốn xóa tất cả những gì đang có trên màn hình ta dùng lệnh

**ClrScr;**

- Nhãn: **Label lệnh1, lệnh2;**
- Hằng: **Const hằng=trị\_hằng;**



- Kiểu dữ liệu tự tạo: **Type <tên\_kiểu> = <kiểu\_dữ\_liệu>;**
  - Biến: **Var tên\_biến:kiểu\_dữ\_liệu;**
- Chú ý: thứ tự khai báo như trên là bắt buộc cho mọi chương trình.

## II/ Phần thân chương trình:

**Begin**  
[<dãy\_lệnh>]

**End.**

## III/ Ví dụ chương trình đơn giản:

1/ Ví dụ 1:

```
program VD1;
begin
  writeln('Hello');
end.
```

2/ Ví dụ 2:

```
program VD2;
uses crt;
begin
  clrscr;
  write('Xin chào các bạn');
  readln;
end.
```

## **BÀI 2: CÁC KIỂU DỮ LIỆU CHUẨN**

### I/ Kiểu nguyên:

Kiểu	Bộ nhớ lưu trữ	Phạm vi
Byte	1 byte	0 đến 255
Integer	2 byte	$-2^{15}$ đến $2^{15}-1$
Word	2 byte	0 đến $2^{16}-1$
Longint	4 byte	$-2^{31}$ đến $2^{31}-1$
Shortint	1 byte	-128 đến 127

- Kiểu Byte và Word không có thuộc tính dấu.

- Các phép toán trên số nguyên:

1/ Các phép toán số học: +, -, \*, /.

2/ Phép chia:

+ Phép chia hai số nguyên cho kết quả là một số thực.

+ Phép chia lấy phần nguyên: **x Div y;**

Vd: 15 Div 2 = 7.

+ Phép chia lấy phần dư: **x Mod y;**

Vd: 15 Mod 2 = 1.

3/ Tính chẵn/lẻ: **Odd(x);**

**True:** x là số lẻ, **False:** x là số chẵn.

4/ Lấy số đứng trước: **Pred(x);** Vd: Pred(2) = 1.

5/ Lấy số đứng sau: **Succ(x);** Vd: Succ(2) = 3.

6/ Phép toán quan hệ: <>, =, >=, <=, >, <.

- Xử lý số nguyên dưới dạng nhị phân:

**Các phép toán lô-gic:**



- + **x And y**: cho giá trị 1 nếu x, y đều là 1.
- + **x Or y**: cho giá trị 0 nếu x, y đều là 0.
- + **not x**: cho giá trị ngược lại giá trị của x.
- + **x Xor y**: cho giá trị 1 nếu x, y có giá trị khác nhau.

Bảng chân trị:

X	Y	X And Y	X Or Y	Not X	Not Y	X Xor Y
0	0	0	0	1	1	0
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	1	0	0	0

- Các phép toán chuyển dịch số học: Cho  $x, n \in \mathbb{N}^+$

- + **x Shl n**: dịch chuyển sang trái số nguyên x đi n bit, bằng phép nhân  $x * 2^n$ .
- + **x Shr n**: dịch chuyển sang phải số nguyên x đi n bit, bằng phép chia  $x \text{ Div } 2^n$ .

## II/ Kiểu thực:

Kiểu	Bộ nhớ lưu trữ	Phạm vi
Single	4 byte	$1,5 \cdot 10^{-45}$ đến $3,4 \cdot 10^{38}$
Real	6 byte	$2,9 \cdot 10^{-39}$ đến $1,7 \cdot 10^{38}$
Double	8 byte	$5 \cdot 10^{-324}$ đến $1,7 \cdot 10^{308}$
Extended	10 byte	$1,9 \cdot 10^{-4951}$ đến $1,1 \cdot 10^{4932}$

1/ Các phép toán số học, quan hệ: (như số nguyên).

2/ Các hàm: Cho  $x \in \mathbb{R}$ :

- **Abs(x)**: giá trị tuyệt đối của x.
- **Sqr(x)**: giá trị bình phương của x.
- **Sin(x), Cos(x), Arctan(x)**: giá trị sin(x), cos(x), arctan(x) (rad).
- **Ln(x)**: giá trị logarit theo cơ số e của x.
- **Exp(x)**: giá trị lũy thừa của x theo cơ số e.
- **Sqrt(x)**: giá trị căn bậc hai của x.
- **Trunc(x)**: giá trị phần nguyên của x.
- **Round(x)**: giá trị phần nguyên được làm tròn của x.

## III/ Kiểu ký tự:

Kiểu	Bộ nhớ lưu trữ	Phạm vi
Char	1 byte	256 ký tự mã ASCII

Các hàm xử lý ký tự:

- **Ord(x)**: trả về mã số ASCII của ký tự x.
- **Chr(n)**: trả về ký tự tương ứng của mã số n trong bảng mã ASCII.
- **Uppcase(x)**: chuyển ký tự thường thành ký tự hoa.
- **Pred(x)**: cho ký tự đứng trước x.
- **Succ(x)**: cho ký tự đứng sau x.

## IV/ Kiểu chuỗi hay xâu ký tự:

Kiểu chuỗi có chiều dài 255. Một chuỗi được viết trong một cặp dấu nháy đơn. Vd: 'Truong THPT Chuyen Hung Vuong – Binh Duong'.

Khai báo: **Var <tên\_biến>:String[độ\_dài\_chuỗi];**

## V/ Kiểu lô-gic:

- Các giá trị: True, False.
- Quan hệ thứ tự: False < True.



- Các phép toán: (giống số nguyên).

### BÀI 3: THỦ TỤC CHUẨN VÀO/RA

#### I/ Thủ tục viết dữ liệu ra màn hình:

- Viết dữ liệu ra màn hình và không có xuống dòng: **Write(Item1, Item2,...);**
- Viết dữ liệu ra màn hình và có xuống dòng: **Writeln(Item1, Item2,...);**

**Chú ý:** Item1, Item2,... có thể là kí tự, chuỗi kí tự, biến, hàm, biểu thức.

#### II/ Thủ tục vào dữ liệu từ bàn phím:

- Vào dữ liệu và không có xuống dòng: **Read(biến 1, biến 2,...);**
- Vào dữ liệu rồi di chuyển con trỏ xuống đầu dòng kế: **Readln(biến 1, biến 2,...);**

#### III/ Phép gán: **tên\_biến:=biểu\_thức;**

**Chú ý:** biến và biểu thức phải cùng kiểu.

#### IV/ Câu lệnh ghép: ghép các lệnh thành một khối:

```
begin
    <câu_lệnh>;
end;
```

#### V/ Định dạng hiển thị:

```
Write(tên_biến, biểu_thức:độ_rộng:số_chữ_số_thập_phân);
Writeln(tên_biến, biểu_thức:độ_rộng:số_chữ_số_thập_phân);
```

##### • Số nguyên:

- Khai báo: **var n:integer;**
- Hiển thị độ rộng của trường ra là 4 và canh phải: **writeln(n:4);**

##### • Số thực:

- Khai báo: **var x:real;**
- Hiển thị độ rộng của trường ra là 8, lấy 2 số lẻ phần thập phân và canh phải: **writeln(x:8:2);**

### *Bài Tập*

1) Viết các biểu thức toán học dưới đây trong Pascal:

- a)  $(x + y)/(x - y)$ ;      b)  $\frac{b^2 - 4ac}{2a}$ ;      c)  $|x + y| + \log_2 x$ ;      d)  $(1 + z) \frac{x + \frac{y}{z}}{a - \frac{1}{1 + x^3}}$ ;      e)  $2\cos x + 3\sin y$

#### **Giải:**

- a)  $(x+y)/(x-y)$ .  
 b)  $(\text{sqr}(x)-4*a*c)/(2*a)$ .  
 c) Ta có:  **$\log_a x = \ln(x)/\ln(a)$** .  
      **$\text{Abs}(x+y) + \ln(x)/\ln(2)$** .  
 d)  $(1+z)*((x+y/z)/(a-1/(1+x*x*x)))$ .  
 e)  $2*\cos(x) + 3*\sin(y)$ .

2) Viết các biểu thức từ dạng toán học sang Pascal:

- a)  $e^y + 5^{(2x+5)} \cos x$ ;      b)  $\sqrt{x + \sqrt{x + \sqrt{x}}}$ ;      c)  $x^3 + \log_2(x^4 + 2) - \sqrt{x - 4}$ ;      d)  $x^2 + y^2 \leq R^2$ ;      e)  $100 \geq N > 0$ ;      f)  $\left| \sin \frac{\pi x}{2} \right| < \frac{1}{2}$

#### **Giải:**





- a)  $\text{Exp}(y) + \exp((2x+5) * \ln(5)) * \cos(x)$ .
- b)  $\text{Sqrt}(x + \text{sqrt}(x + \text{sqrt}(x)))$ .
- c)  $\text{Exp}(3 * \ln(x)) + \ln(\exp(4 * \ln(x)) + 2) / \ln(2) - \text{sqrt}(x-4)$ .
- d)  $\text{Sqr}(x) + \text{sqr}(y) \leq \text{sqr}(R)$ .
- e)  $(0 < N) \text{ and } (N \leq 100)$ .
- f)  $\text{Abs}(\sin((3.14 * x) / 2)) < 1/2$ .

3) Chuyển các biểu thức sau sang Pascal:

- a)  $\frac{a + \sin x}{\sqrt{a^2 + x^2 + 1}}$  ;
- b)  $\frac{2e^{x+y}}{x^2 + y^2}$  ;
- c)  $\frac{\sin^2 x}{y + 0.5}$  ;
- d)  $\frac{\sqrt[3]{-8}}{1 + e^x}$

**Giải:**

- a)  $(a + \sin(x)) / \text{sqrt}(a * a + x * x + 1)$ .
- b)  $(2 * \exp(x + y)) / (\text{sqr}(x) * \text{sqr}(y))$ .
- c)  $\text{Sqr}(\sin(x)) / (y + 0.5)$ .
- d)  $\text{Exp}(1/3 * \ln(-8)) / (1 + \exp(x))$ .

4) Chuyển các biểu thức Pascal sang dạng toán học:

- a)  $\text{Sqrt}(p * (p-a) * (p-b) * (p-c))$ .
- b)  $\text{Abs}(x-y) / (x * x + \text{sqr}(y) + 1)$ .
- c)  $\cos(3 * \pi * x / 2) + \sin(3 * \pi * x / 2)$ .
- d)  $a/b/c/d$ .

**Giải:**

- a)  $\sqrt{p(p-a)(p-b)(p-c)}$  ;
- b)  $\frac{|x-y|}{x^2 + y^2 + 1}$  ;
- c)  $\cos\left(3\pi \frac{x}{2}\right) + \sin\left(3\pi \frac{x}{2}\right)$  ;
- d)  $\frac{a}{\frac{b}{\frac{c}{d}}}$

5) Lập trình nhập từ bàn phím các số thực a, b, c, d và x. Tính và đưa ra màn hình giá trị biểu thức  $ax^3 + bx^2 + cx + d$ .

**Giải:**

```

program baitap;
uses crt;
var a, b, c, d, x, kq:real;
begin
  clrscr;
  writeln('Nhap vao a, b, c, d, x: ');
  readln(a, b, c, d, x);
  kq := a * x * x * x + b * x * x + c * x + d;
  writeln('Ket qua la: ', kq:8:2);
  readln;
end.

```

6) Một người đi xe đạp với tốc độ 10km/h và một người đi xe máy với tốc độ 30km/h cùng xuất phát từ một vị trí, cùng một thời điểm và đi cùng một hướng. Lập trình tính khoảng cách giữa hai người sau t giờ?

**Giải:**

```

program baitap;

```



```

uses crt;
var t:integer;
begin
  clrscr;
  write('t= '); readln(t);
  writeln('Khoang cach la: ',20*t,' km');
  readln;
end.

```

7) Lập trình nhập từ bàn phím tọa độ ba đỉnh A, B, C của tam giác ABC.

- Tính và đưa ra diện tích tam giác theo công thức Hê-rông?
- Tính và đưa ra độ dài các đường cao  $h_a$ ,  $h_b$ ,  $h_c$ ?

**Giải:**

```

program baitap;
uses crt;
var xa,ya,xb,yb,xc,yc:real;
    p,s,ha,hb,hc:real;
    a,b,c:real;
begin
  clrscr;
  write('Toa do diem A: '); readln(xa,ya);
  write('Toa do diem B: '); readln(xb,yb);
  write('Toa do diem C: '); readln(xc,yc);
  a:=sqrt(sqr(xb-xc)+sqr(yb-yc));
  b:=sqrt(sqr(xa-xc)+sqr(ya-yc));
  c:=sqrt(sqr(xb-xa)+sqr(yb-ya));
  p:=(a+b+c)/2; s:=sqrt(p*(p-a)*(p-b)*(p-c));
  ha:=2*s/a; hb:=2*s/b; hc:=2*s/c;
  writeln('Dien tich tam giac = ',s:8:2);
  writeln('ha= ',ha:8:2,' hb= ',hb:8:2,' hc= ',hc:8:2);
  readln;
end.

```

8) Lập trình nhập từ bàn phím hai số thực a và b, tính và đưa ra màn hình:

- Trung bình cộng các bình phương của hai số đó?
- Trung bình cộng các giá trị tuyệt đối của a và b?

**Giải:**

```

program baitap;
uses crt;
var a,b,kq:real;
begin
  clrscr;
  write('Nhap a, b = ');
  readln(a,b);
  kq:=(sqr(a)+sqr(b))/2;
  writeln('Trung binh cong cac binh phuong: ',kq:8:2);

```



```
kq:=(abs(a)+abs(b))/2;
writeln('Trung binh cong cac gia tri tuyet doi: ',kq:8:2);
readln;
```

end.

- 9) Lập trình nhập từ bàn phím hai số nguyên khác 0 M và N ( $M, N \leq 32767$ ). Tính tổng, hiệu, tích, thương của hai số đó và đưa ra màn hình dưới dạng: <biểu thức số học>=<kết quả>.
- 10) Lập trình và đưa ra màn hình diện tích hình vành khuyên có bán kính ngoài là R1 và bán kính trong là R2 ( $R1, R2 \in \mathbb{R}, 0 < R1 < R2 < 10^5$ ). Kết quả được đưa ra có độ chính xác với bốn chữ số thập phân.
- 11) Cho  $\Delta ABC$  đều cạnh a ( $a \in \mathbb{R}, 0 < a < 200$ ). Tính diện tích hình vành khuyên tạo bởi đường tròn ngoại tiếp và đường tròn nội tiếp  $\Delta ABC$ .

**Giải:**

```
program baitap;
uses crt;
const pi=3.14;
var a,h,rngt,rnt:real;
    s,sngt,snt:real;
begin
    clrscr;
    writeln('Nhap canh cua tam giac deu: ');
    readln(a);
    h:=a*sin(pi/3);
    rnt:=h/3;
    rngt:=2*rnt;
    sngt:=rngt*rngt*pi;    snt:=rnt*rnt*pi;
    s:=sngt-snt;
    writeln('Dien tich hinh vanh khuyen can tim la: ',s:8:2);
    readln;
end.
```

- 12) Viết chương trình giải phương trình bậc hai (trường hợp có hai nghiệm phân biệt).

**Giải:**

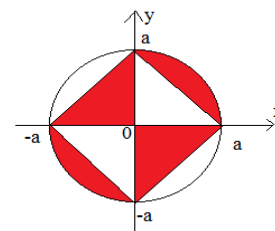
```
program baitap;
uses crt;
var a,b,c,d,x1,x2:real;
begin
    clrscr;
    write('Nhap a, b, c = ');
    readln(a,b,c);
    d:=sqr(b)-4*a*c;
    x1:=(-b-sqrt(d))/(2*a);
```



```
x2:=-b/a-x1;
writeln('x1 = ',x1:8:2,' x2 = ',x2:8:2);
readln;
```

end.

- 13)Viết chương trình nhập vào số a (a>0) rồi tính diện tích phần tô màu trong hình sau:



- 14)Lập trình tính và đưa ra màn hình vận tốc khi chạm đất của một vật rơi tự do từ độ cao h, lấy  $g = 10 \text{ m/s}^2$ ?

- 15)Viết chương trình nhập vào một kí tự hoa sau đó hiển thị dạng chữ thường của nó?

**Giải:**

```
program baitap;
uses crt;
var ch:char;
begin
    clrscr;
    write('Nhap vao chu hoa: ');
    readln(ch);
    writeln('Chu thuong tuong ung la: ',chr(ord(ch)+32));
    readln;
end.
```

➤ **Chú ý:** chr(n) = #n.

- 16)Viết chương trình tính chu vi và diện tích của hình chữ nhật, hình vuông, hình tròn, tam giác vuông (thường)?

- 17)Viết chương trình chuyển đổi số nguyên (đơn vị giây) sang dạng: ... giờ ... phút ... giây?

**Giải:**

```
program baitap;
uses crt;
var gio,phut,giaiy:integer;
begin
    clrscr;
    write('Nhap vao so giay: ');
    readln(giaiy);
    gio:=giaiy div 3600;
    phut:=giaiy mod 3600 div 60;
    eritln(giaiy,' giay = ',gio,' gio ',phut,' phut ',giaiy mod 360 mod 60,' giay');
    readln;
end.
```

- 18)Viết chương trình tính và đưa ra giá trị biểu thức:

$$a) A = \frac{2e^{x+y}}{x^3+y^2}; \quad b) B = \frac{\sqrt{8}}{1-e^x}; \quad c) 2 \cos x + 3 \sin y; \quad d) \frac{b^2-4ac}{2a}$$

$$e) |x+y| + \log_2 x; \quad f) (1+z) \frac{x+\frac{y}{z}}{1+x^3}; \quad g) \frac{\sqrt[3]{-8}}{1+e^x}$$

- 19)Viết chương trình nhập vào một số nguyên có 3 chữ số, sau đó in ra tổng các chữ số?



20)Viết chương trình nhập vào số nguyên có 3 chữ số, sau đó in ra số ngược lại?

**Giải:**

```
program baitap;
uses crt;
var n,a,b,c,s:integer;
begin
    clrscr;
    write('Nhap vao so nguyen: ');
    readln(n);
    a:=n div 100;
    b:=(n mod 100) div 10;
    c:=n mod 10;
    s:=c*100+b*10+a;
    writeln('So nguoc lai voi 'n,' la: ',s);
    readln;
end.
```

21)Viết chương trình hoán vị hai số nguyên a và b? (Gợi ý: sử dụng biến tạm)

22)Viết chương trình in ra nhãn của một quyển tập theo mẫu:

*Truong: .....  
 Lop: ..... Tap: .....  
 Ten: .....  
 Nien khoa: 20... - 20...*

23)Viết chương trình nhập vào 2 số nguyên và tính giá trị biểu thức  $a^b$ . Tương tự:  $(a^b)^c$ ,  $[(a^b)^c]^d$ ?  
 (Gợi ý:  $a^b = \exp(b \cdot \ln(a))$  với  $a > 0$ ).

24)Nhập vào số giờ, phút, giây, sau đó nhập thêm số giây và cho biết số giờ, phút, giây mới?

**Giải:**

```
program baitap;
uses crt;
var gio,phut,giay:integer;
    kq1,kq2,them:integer;
begin
    clrscr;
    write('Nhap so gio: '); readln(gio);
    write('Nhap so phut: '); readln(phut);
    write('Nhap so giây: '); readln(giay);
    kq1:=gio*3600+phut*60+giay;
    write('Nhap so giây them: '); readln(them);
    kq2:=kq1+them;
    writeln('So gio moi la: ',kq2 div 3600);
    writeln('So phut moi la: ',kq2 mod 3600 div 60);
    writeln('So giây moi la: ',kq2 mod 3600 mod 60);
    readln;
end.
```

25)Viết chương trình đổi độ sang radian và ngược lại, biết:



$$\text{radian} = \frac{\text{độ} \cdot \pi}{180} \quad \text{độ} = \frac{\text{radian} \cdot 180}{\pi}$$

26) Nhập vào số tiền muốn đổi hoặc rút ra từ ngân hàng. Cho biết số lượng giấy bạc có mệnh giá 100000, 50000, 20000, 10000, 5000, 1000 được nhận sao cho số lượng giấy bạc có mệnh giá cao nhất được đổi nhiều nhất?

Giải:

```

program baitap;
uses crt;
var n:longint;
    kq1,kq2,kq3,kq4,kq5,kq6,kq7,kq8:integer;
Begin
    clrscr;
    writeln('Nhap vao so tien can doi: ');
    readln(n);
    kq1:=n div 100000;   kq2:=n mod 100000;   kq3:=kq2 div 50000;
    kq4:=kq2 mod 50000 div 20000;   kq5:=kq2 mod 50000 mod 20000;
    kq6:=kq5 div 10000;   kq7:=kq5 mod 10000 div 5000;
    kq8:=kq5 mod 10000 mod 5000;
    writeln('So to giay bac 100000 VND la: ',kq1);
    writeln('So to giay bac 50000 VND la: ',kq3);
    writeln('So to giay bac 20000 VND la: ',kq4);
    writeln('So to giay bac 10000 VND la: ',kq6);
    writeln('So to giay bac 5000 VND la: ',kq7);
    writeln('So to giay bac 1000 VND la: ',kq8);
    readln;
end.

```

### ***ĐỌC THÊM:* Trang trí màu nền và màu chữ trong Pascal:**

- Khai báo: uses crt;
- Các thủ tục trình bày màn hình:
  - Di chuyển con trỏ đến cột X(1-80) và dòng Y(1-25): *GotoXY(X,Y);*
  - Xoá màn hình: *Clrscr;*
  - Xóa từ con trỏ đến hết dòng: *ClrEol;*
  - Thiết lập màu cho kí tự: *TextColor(tên màu hoặc mã màu);*
  - Thiết lập màu cho nền của màn hình: *TextBackGround(tên màu hoặc mã màu);*
  - Mã màu và tên màu:

<i>Mã màu</i>	<i>Tên màu</i>	<i>Mã màu</i>	<i>Tên màu</i>
<b>0</b>	Black	<b>8</b>	Dark Grey
<b>1</b>	Blue	<b>9</b>	Light Blue
<b>2</b>	Green	<b>10</b>	Light Green
<b>3</b>	Cyan	<b>11</b>	Light Cyan
<b>4</b>	Red	<b>12</b>	Light Red
<b>5</b>	Magenta	<b>13</b>	Light Magenta



6	Brown	14	Yellow
7	Light Grey	15	White

## CHƯƠNG III. CẤU TRÚC Rẽ NHÁNH VÀ LẶP

### BÀI 1: CẤU TRÚC Rẽ NHÁNH

#### I/ Rẽ nhánh dạng thiếu:

If <điều kiện> Then <câu lệnh>;

Nếu điều kiện có giá trị true thì thực hiện câu lệnh

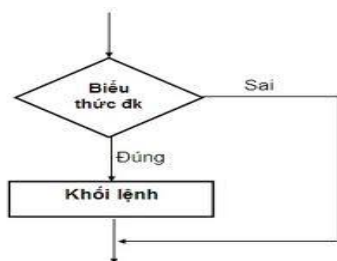
#### II/ Rẽ nhánh dạng đầy đủ:

If <điều kiện> Then <câu lệnh 1> Else <câu lệnh 2>;

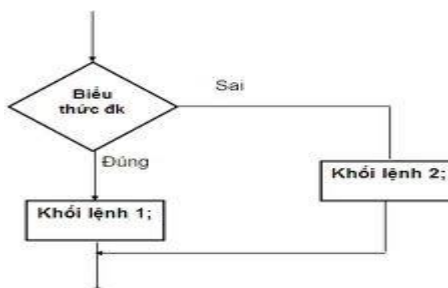
Nếu điều kiện có giá trị là true thì thực hiện câu lệnh 1, ngược lại nếu điều kiện có giá trị là false thì thực hiện câu lệnh 2.

Lưu ý: Trước Else không có dấu chấm phẩy “;”.

#### III/ Lưu đồ hoạt động:



Dạng 1



Dạng 2

### BÀI 2: CẤU TRÚC LẶP

#### I/ Lặp với số lần biết trước:

- Có số lần lặp xác định trước.
- Sau khi thực hiện thân vòng lặp tự động tăng (giảm) biến đếm một đơn vị.
- Cú pháp:

Dạng 1: Biến đếm tăng dần (dạng lặp tiến):

For <biến đếm>:=<giá trị đầu> To <giá trị cuối> Do <câu lệnh>;

Dạng 2: Biến đếm giảm dần (dạng lặp lùi):

For <biến đếm>:=<giá trị cuối> Downto <giá trị đầu> Do <câu lệnh>;

- Biến đếm là biến đơn, có kiểu nguyên.
- Giá trị đầu, giá trị cuối là các biểu thức cùng kiểu với biến đếm và giá trị đầu phải nhỏ hơn hoặc bằng giá trị cuối.

#### II/ Lặp với số lần chưa biết trước:

- Có số lần lặp không xác định.
- Kiểm tra điều kiện tiếp tục vòng lặp ở đầu vòng lặp.

Cú pháp: While <điều kiện> Do <câu lệnh>;

- Điều kiện là biểu thức lô-gic.
- Nếu điều kiện có giá trị True thì thực hiện thân vòng lặp và ngược lại.
- Câu lệnh là một câu lệnh đơn hoặc ghép.

#### III/ Vòng lặp Repeat...Until....:

- Có số lần lặp không xác định.
- Kiểm tra điều kiện dừng vòng lặp ở cuối vòng lặp.



Cú pháp: **Repeat** <câu lệnh> **Until** <điều kiện>;

- Điều kiện có giá trị kiểu lô-gic boolean (True/False).

- Nếu điều kiện có giá trị False thì thực hiện lại thân vòng lặp và ngược lại.

### BÀI 3: CÂU LỆNH LỰA CHỌN CASE...OF...

#### I/ Dạng 1 (không có Else):

**Case** <biểu thức> **Of**

<hằng 1> : <câu lệnh 1>;

<hằng 2> : <câu lệnh 2>;

[...]

<hằng n> : <câu lệnh n>;

**End;**

#### II/ Dạng 2 (có Else):

**Case** <biểu thức> **Of**

<hằng 1> : <câu lệnh 1>;

<hằng 2> : <câu lệnh 2>;

[...]

<hằng n> : <câu lệnh n>

**Else** <câu lệnh n+1>;

**End;**

**Lưu ý:** khi có nhiều giá trị của <biểu thức> cho phép thực hiện cùng một câu lệnh thì có thể sắp xếp các giá trị đó trên cùng một hàng, ngăn cách nhau bởi dấu phẩy và chỉ viết câu lệnh một lần mà thôi thay cho việc phải viết nhiều lần. Ví dụ: <hằng 1>,<hằng 2>,...,<hằng n>:<câu lệnh>;

### BÀI TẬP

Bài 1: Viết chương trình nhập vào ba số và cho biết nó có phải là bộ số Pythagore không?

Giải:

```
program bt;
uses crt;
var a,b,c:integer; a2,b2,c2:longint;
begin
  clrscr;
  write('Nhập a,b,c = '); readln(a,b,c);
  a2:=a; b2:=b; c2:=c;
  a2:=a2*a; b2:=b2*b; c2:=c2*c;
  if (a2=b2+c2) or (b2=a2+c2) or (c2=a2+b2) then writeln('La bo so Pithagore')
  else writeln('Khong phai la bo so Pithagore');
  readln;
end.
```

Bài 2: Viết câu lệnh rẽ nhánh và tính:

$$z = \begin{cases} x^2 + y^2, & \text{nếu } x^2 + y^2 \leq 1 \\ x + y, & \text{nếu } x^2 + y^2 > 1 \text{ và } y \geq x \\ 0,5, & \text{nếu } x^2 + y^2 > 1 \text{ và } y < x \end{cases}$$

Giải:

```
program Bt;
```





```

uses crt;
var x,y,z:integer;
begin
  clrscr;
  write('Nhap x,y = '); readln(x,y);
  if (x*x+y*y<=1) then z:=x*x+y*y;
  if (x*x+y*y>1) and (y>=x) then z:=x+y;
  if (x*x+y*y>1) and (y<x) then z:=0.5;
  writeln('z = ',z);
  readln;
end.

```

Bài 3: Lập trình tính:

a)  $X = \sum_{n=1}^{30} \frac{1}{n}$ ;      b)  $Y = \sum_{n=1}^{50} \frac{n}{n+1}$ ;

c)  $Z = \frac{1}{2.3} + \frac{2}{3.4} + \frac{3}{4.5} + \dots + \frac{n-1}{n.(n+1)}$  với n là số lớn nhất thoả  $\frac{n-1}{n.(n+1)} > 2.10^{-2}$ .

Giải:

a)

```

program Bt;
uses crt;
var x:real; n:integer;
begin
  clrscr; x:=0;
  for n:=1 to 30 do x:=x+1/n;
  writeln('X = ',x:5:2);
  readln;
end.

```

b) (Gợi ý)

```

y:=0;
for n:=1 to 50 do y:=y+n/(n+1);

```

c) (Gợi ý)

```

z:=0; n:=2;
while ((n-1)/n/(n+1)>2E-2) do
begin
  z:=Z+(n-1)/n/(n+1);
  n:=n+1;
end;

```

Bài 4: Nhập vào ba số nguyên, tìm số lớn nhất?

Giải:

```

program bt;
uses crt;
var a,b,c,max:integer;
begin
  clrscr;
  write('Nhap vao ba so nguyen: '); readln(a,b,c);
  max:=a;

```



```

    if max < b then max := b;
    if max < c then max := c;
    writeln('Max = ', max);
    readln;
end.

```

Bài 5: Viết chương trình nhập vào ba số dương, kiểm tra xem chúng có phải là ba cạnh của tam giác không? Nếu có thì cho biết đó là tam giác gì?

Giải:

```

program bt;
uses crt;
var a, b, c: integer;
begin
    clrscr;
    write('Nhập vào a, b, c = '); readln(a, b, c);
    if (a + b > c) and (b + c > a) and (c + a > b) then
        begin
            write('Đây là một tam giác ');
            if (a = b) and (b = c) then writeln('đều') else
                if (a = b) or (b = c) or (c = a) then writeln('cân') else
                    if (a*a + b*b = c*c) or (a*a + c*c = b*b) or (b*b + c*c = a*a) then writeln('vuông') else
                        writeln('thường');
        end else
            writeln('Không thể tạo thành tam giác!');
    readln;
end.

```

Bài 6: Viết chương trình xếp loại học tập của học sinh dựa vào điểm trung bình như sau:  
 + ĐTB  $\geq 8.0$ : giỏi; +  $6.5 \leq \text{ĐTB} \leq 8.0$ : khá; +  $5.0 \leq \text{ĐTB} \leq 6.5$ : trung bình; + ĐTB  $\leq 5.0$ : yếu.

Giải:

```

program bt;
uses crt;
var dtb: real;
begin
    clrscr;
    write('Nhập vào ĐTB: '); readln(dtb);
    if dtb >= 8.0 then writeln('Xếp loại giỏi') else
        if dtb >= 6.5 then writeln('Xếp loại khá') else
            if dtb >= 5.0 then writeln('Xếp loại trung bình') else writeln('Xếp loại yếu');
    readln;
end.

```

Bài 7: Lập trình để giải bài toán cổ sau:

*“Vừa gà vừa chó,  
 Bó lại cho tròn,  
 Ba mươi sáu con,  
 Một trăm chân chẵn”*



Hỏi có bao nhiêu con mỗi loại?

Giải:

```

program bt;
uses crt;
var ga,cho:integer;
begin
  clrscr;
  for ga:=1 to 35 do
    for cho:=1 to 35 do
      if (ga+cho=36) and (2*ga+4*cho=100) then
        begin
          writeln('So ga la: ',ga,' con');
          writeln('So cho la: ',cho,' con');
        end;
      readln;
    end.

```

Bài 8: Lập trình để giải phương trình bậc nhất  $ax + b = 0$ , có các trường hợp sau:

+  $a \neq 0$ : phương trình có 1 nghiệm  $x = -b/a$ .

+  $a = 0$  và  $b \neq 0$ : phương trình vô nghiệm.

+  $a = 0$  và  $b = 0$ : phương trình vô định.

Giải:

```

program bt;
uses crt;
var a,b:real;
begin
  clrscr;
  write('a, b = '); readln(a,b);
  if a < > 0 then writeln('Nghiem x = ', -b/a:5:2) else
    if b < > 0 then writeln('Phuong trinh vo nghiem') else
      writeln('Phuong trinh vo dinh');
  readln;
end.

```

Bài 9: Lập trình giải phương trình bậc hai  $ax^2 + bx + c = 0$  ?

Giải:

```

program bt;
uses crt;
var a,b,c,d,x1,x2:real;
begin
  clrscr;
  write('a, b, c = '); readln(a,b,c);
  d:=sqr(b)-4*a*c;
  if d>0 then
    begin
      x1:=(-b+sqrt(d))/(2*a);
      x2:=(-b-sqrt(d))/(2*a);
    end

```



```

        writeln('x1 = ',x1:5:2,' x2 = ',x2:5:2);
    end else
        if d=0 then writeln('x = ',-b/(2*a):5:2) else
            writeln('Phuong trinh vo nghiem');
        readln;
    end.

```

Bài 10: Lập trình để giải hệ phương trình tuyến tính sau:  $\begin{cases} ax + by = c \\ dx + ey = f \end{cases} (a, b, c, d, e, f \neq 0)$

Giải:

```

program Bt;
uses crt;
var a,b,c,d,e,f,d1,dx,dy:real;
begin
    clrscr;
    write('a, b, c = '); readln(a,b,c);
    write('d, e, f = '); readln(d,e,f);
    d1:=a*e-b*d;
    dx:=c*e-b*f;
    dy:=a*f-c*d;
    if (d1=0) and (dx=0) then writeln('Hpt vo dinh') else
        if (d1=0) and (dx<>0) then writeln('Hpt vo nghiem') else
            if (d1<>0) then writeln('x = ',dx/d1:5:2,' ',y = ',dy/d1:5:2);
        readln;
    end.

```

Bài 11: Viết chương trình nhập một kí tự, nếu kí tự nhập vào là kí tự thường thì in ra kí tự hoa tương ứng và ngược lại?

Giải:

```

program bt;
uses crt;
var ch:char;
begin
    clrscr;
    write('Nhap ki tu: '); readln(ch);
    if (ord(ch) >= 41) and (ord(ch) <= 90) then writeln(chr(ord(ch) + 32)) else
        if (ord(ch) >= 97) and (ord(ch) <= 122) then writeln(chr(ord(ch)-32)) else
            writeln('Ki tu ban nhap khong hop le!!!');
        readln;
    end.

```

Bài 12: Viết chương trình tính tổng 100 số tự nhiên đầu tiên?

Giải:

```

uses crt;
var i:integer; s:longint;
begin
    clrscr;          s:=0;
    for i:=1 to 100 do s:=s+i;

```



```
writeln('Sum = ',s);
readln;
```

end.

Bài 13: Viết chương trình tính các tổng sau:

- |                                  |  |
|----------------------------------|--|
| a) $1 + 3 + 5 + \dots + n$       | b) $2 + 4 + 6 + \dots + n$                               |
| c) $1 + 2^2 + 3^2 + \dots + n^2$ | d) $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$ |

Hướng dẫn:

S:=0;

a) for i:=1 to n do if (i mod 2=1) then s:=s+i;

b) for i:=1 to n do if (i mod 2=0) then s:=s+i;

c) for i:=1 to n do s:=s+sqr(i);

d) for i:=1 to n do s:=s+1/n;

writeln('Sum = ',s);

Bài 14: Viết chương trình in ra bảng nhân n, với n được nhập từ bàn phím?

Hướng dẫn:

```
for i:=1 to 10 do
begin
    for j:=2 to 9 do write(j:2, ' x ',i:2, ' = ',j*i:2);
    writeln;
end;
```

Bài 15: Một người gửi tiền ngân hàng với số tiền ban đầu là A (đơn vị: triệu đồng), lãi suất mỗi tháng là k%. Hãy cho biết số tiền người đó rút ra sau khoảng thời gian t tháng biết rằng tiền lãi được tính lãi lũy kế theo chu kỳ c tháng, nghĩa là sau mỗi tháng lãi suất sẽ được cộng vào tiền gốc nhưng đúng chu kỳ c tháng mới cho phép được rút?

Giải:

```
program bt;
uses crt;
var tiengoi,laisuat,tienco,y:real; thang,chuky,i:integer;
begin
    clrscr;
    write('Nhap so tien goi: '); readln(tiengoi);
    write('Nhap he so lai suat: '); readln(laisuat);
    write('Nhap so thang: '); readln(thang);
    write('Nhap chu ki: '); readln(chuky);
    tienco:=tiengoi;
    for i:=1 to thang do
    begin
        tienco:=tienco+tienco*laisuat/100;
        if (i mod chuky=0) then y:=tienco;
    end;
    writeln('So tien nhan duoc la: ',y);
    readln;
end.
```



Bài 16: Viết chương trình đưa ra màn hình ước số chung lớn nhất của hai số nguyên dương a và b (sử dụng thuật toán Euclide), với a và b được nhập từ bàn phím?

Giải:

```

program bt;
uses crt;
var a,b,r:integer;
begin
    clrscr;
    write('a, b = '); readln(a,b);
    r:=a mod b;
    while r<>0 do
        begin
            a:=b;
            b:=r;
            r:=a mod b;
        end;
    writeln('UCLN = ',b);
    readln;
end.

```

Bài 17: Viết chương trình tính bội chung nhỏ nhất của hai số nguyên dương a và b được nhập từ bàn phím?

Hướng dẫn:  $BCNN(a, b) = \frac{ab}{UCLN(a,b)}$

Bài 18: Viết chương trình kiểm tra một số n được nhập từ bàn phím có phải là số nguyên tố hay không?

Giải:

```

program bt;
uses crt;
var n,i:integer;
begin
    clrscr;
    write('Nhap n = '); readln(n);
    if n<2 then writeln(n,' khong la so nguyen to') else
        for i:=2 to trunc(sqrt(n)) do
            if n mod i=0 then writeln(n,' khong la so nguyen to') else
                writeln(n,' la so nguyen to');
    readln;
end.

```

Bài 19: Năm 1996 dân số nước ta là A = 75 triệu người, tốc độ tăng dân số là k% một năm. Lập trình in dân số của từng năm cho đến khi dân số tăng s lần so với năm 1996. Chạy chương trình với k = 2,3%; s = 1,5 lần.

Giải:

```

program bt;
uses crt;
const k=0.023; s=1.5; a=75000000;

```



```

var tl,tsd,sdt:real; i:integer;
begin
    clrscr;
    writeln('Ket qua so dan tung nam: ');
    tl:=1; tsd:=a; i:=1;
    while tl<=s do
        begin
            sdt:=tsd*k;
            tsd:=tsd+sdt;
            tl:=tsd/a;
            writeln('So dan nam thu ',i,' la: ',tsd:12:2);
            inc(i);
        end;
    writeln('Da dat den khi dan so tang gap ',s:0:1,' lan so voi nam 1996');
    readln;
end.

```

Bài 20: Nhập từ bàn phím tuổi của cha và con biết rằng tuổi cha lớn hơn 3 lần tuổi con và tuổi cha và tuổi con phải cùng chẵn hoặc cùng lẻ. Hãy lập trình để trả lời câu hỏi: “Sau bao nhiêu năm nữa thì tuổi cha gấp ba lần tuổi con?”.

Giải:

```

program bt;
uses crt;
var tc,tco,n:word;
begin
    clrscr;
    write('Nhap tuoi cha va con: '); readln(tc,tco); n:=0;
    while tc>3*tco do
        begin
            inc(tc); inc(tco); inc(n);
        end;
    writeln('Sau ',n,' nam nua thi tuoi cha se gap ba lan tuoi con');
    readln;
end.

```

Bài 21: Cho hàm:  $f(x) = \begin{cases} 4x^2 - 5x + 1 & \text{nếu } 0 \leq x \leq 1 \\ \frac{|x-2|}{3(x^2+1)} & \text{nếu } x \leq 0 \text{ hoặc } x > 1 \end{cases}$ . Viết chương trình đọc vào một số thực x từ bàn phím và in ra màn hình giá trị của f(x).

Bài 22: Viết chương trình nhập vào 3 số a, b, c, sau đó in ra màn hình theo thứ tự tăng dần.

Hướng dẫn:

```

tmp:=a;
if tmp<b then tmp:=b;
if tmp<c then tmp:=c;
writeln(tmp);
if a>b then
    begin

```



```

tmp:=a;
a:=b;
b:=tmp;

end;

```

```

if c>a then writeln(c,'a',b) else
if c<b then writeln(a,b,c);

```

Bài 23: Lập trình nhập từ bàn phím ba số nguyên a, b, c. Sau đó kiểm tra 3 số này (theo thứ tự nhập vào) có tạo thành một cấp số cộng hay không rồi đưa kết quả ra màn hình?

Hướng dẫn:

```

if a+c=2*b then writeln('Ba so tao thanh mot cap so cong') else
writeln('Ba so khong tao thanh mot cap so cong');

```

Bài 24: Tương tự bài 23: kiểm tra xem ba số trên có tạo thành một cấp số nhân hay không???

Bài 25: Viết chương trình tính tiền giờ hát karaoke theo qui định trong một ngày từ 6 giờ đến 18 giờ như sau: nếu hát từ 6 giờ đến 12 giờ thì mỗi giờ 20000 đồng. Từ 12 giờ đến 18 giờ mỗi giờ 30000 đồng (giả sử giờ hát và kết thúc đều là các số nguyên)?

Hướng dẫn:

```

if gv<=12 then st:=(gv-gh)*20000 else
if gh>12 then st:=(gv-gh)*30000 else
st:=(12-gh)*20000+(gv-12)*30000;

writeln('So tien phai tra la: ',st);

```

Bài 26: Viết chương trình đọc một số nguyên gồm 4 chữ số (số tạo bởi hai chữ số đầu nhỏ hơn 24, số tạo bởi hai chữ số sau nhỏ hơn 60), rồi hiển thị giờ tương ứng trong ngày. Ví dụ: 0915 -> "9:15 AM", 1320 -> "1:20 PM".

Hướng dẫn:

```

gi:=n div 100;
ph:=n mod 100;
if gi>12 then writeln(gi-12,':',ph,' PM') else
writeln(gi,':',ph,' AM');

```

Bài 27: Một đường thẳng có thể biểu diễn bởi phương trình  $ax + by + c = 0$  ( $a, b, c \in R$ ). Viết chương trình đọc hai bộ số thực trên hai hàng, tương ứng với hai phương trình của hai đường thẳng, sau đó in ra thông báo về quan hệ của hai đường thẳng đó.

Giải:

```

program bt;
uses crt;
var a,b,c,m,n,p:real; ss,tr,vg:boolean;
begin
write('Nhap vao ba he so cua duong thang thu nhat: '); readln(a,b,c);
write('Nhap vao ba he so cua duong thang thu hai: '); readln(m,n,p);
tr:=(a*n=b*m) and (a*p=c*m);
ss:=(a*n=b*m) and (a*p<>c*m);
vg:=(a*m+b*n=0);
if tr then writeln('Hai duong thang trung nhau') else
if ss then writeln('hai duong thang song song') else
if vg then writeln('hai duong thang vuong goc') else
writeln('Hai duong thang cat nhau');

```





```
readln;
```

```
end.
```

📖 Bài 28: In ra bảng sau:

1	2	3	4	5	...	10
11	12	...				20
...						
91	92	...				100

Hướng dẫn:

```
for i:=1 to 100 do
```

```
if i mod 10 < > 0 then write(i:4) else writeln(i:4);
```

📖 Bài 29: Viết chương trình in ra màn hình các kí tự mở rộng trong bộ mã ASCII, biết rằng các kí tự mở rộng là các kí tự có mã ASCII từ 128 đến 255.

Hướng dẫn:

```
for i:=128 to 255 do write(chr(i):2);
```

📖 Bài 30: Viết chương trình tìm và xuất ra màn hình các số nằm trong giữa hai số nguyên a và b cho trước ( $a < b$ ) sao cho các số đó là số chẵn và đồng thời chia hết cho 3?

Hướng dẫn:

```
writeln('Cac so can tim la:');
```

```
for i:=a to b do
```

```
if (i mod 6=0) then write(i, ' ');
```

📖 Bài 31: Tìm một số có ba chữ số sao cho số đó bằng n lần tổng các chữ số của nó?

Hướng dẫn:

```
for a:=1 to 9 do
```

```
for b:=0 to 9 do
```

```
for c:=0 to 9 do
```

```
if 100*a+10*b+c=n*(a+b+c) then writeln(a,b,c);
```

📖 Bài 32: Viết chương trình tính biểu thức:  $\sqrt{x + \sqrt{x + \dots + \sqrt{x}}}$  (n lần x). Trong đó n và x được nhập từ bàn phím và  $n < 100$ .

Hướng dẫn:

```
m:=sqrt(x);
```

```
for i:=1 to n do m:=m+sqrt(x+m);
```

📖 Bài 33: Số hoàn thiện (perfect number) là số tự nhiên có tổng các ước số (kể cả số 1) bằng chính nó. Ví dụ:  $6 = 1 + 2 + 3$ . Viết chương trình hiển thị các số hoàn thiện nhỏ hơn 1000.

Giải:

```
uses crt;
```

```
var n,i,j,s:integer;
```

```
begin
```

```
clrscr;
```

```
write('N = '); readln(n);
```

```
writeln('Cac so hoan thien nho hon ',n,' la: ');
```

```
for i:=2 to n do
```

```
begin
```

```
s:=0;
```

```
for j:=1 to (i div 2) do
```



```

        if i mod j=0 then s:=s+j;
        if s=i then write(i,' ');
    end;
readln;
end.

```

📖 Bài 34: Tìm số nguyên lớn nhất thỏa mãn các điều kiện:

$a) 3n^5 - 317n < 5;$        $b) 6n - 21\sqrt{n} - 1 < 0;$   
 $c) e^n - 2009 \log n < 0;$        $d) 2n^3 - n^2 - 1 > 0.$

Hướng dẫn:

```

n:=1;
a) while 3*sqr(n)*(sqr(n)*n)-317*n<5 do n:=n+1;
b) while 6*n-21*sqr(n)-1<0 do inc(n);
c) while exp(n*ln(n))+2009*ln(n)/ln(10)<0 do inc(n);
d) while 2*n*sqr(n)-sqr(n)-1>0 do inc(n);
writeln('N max = ',n);

```

📖 Bài 35: Viết chương trình nhập liên tục các số nguyên n cho đến khi nhập số 0 thì dừng và in ra:

- Có bao nhiêu số nguyên được nhập?
- Số lớn nhất, số nhỏ nhất?
- Tổng các số đã nhập?

Giải:

```

uses crt;
var dem,max,min,tong,n:integer;
begin
    clrscr;
    write('n = '); readln(n);
    dem:=0; tong:=0;
    while n<>0 do
        begin
            inc(dem); tong:=tong+n;
            if dem=1 then
                begin
                    max:=n; min:=n;
                end else
                begin
                    if n>max then max:=n;
                    if n<=min then min:=n;
                end;
            write('n = '); readln(n);
        end;
    if dem<>0 then
        begin
            writeln('Co tat ca ',dem,' so');
            writeln('So lon nhat la: ',max);
            writeln('So nho nhat la: ',min);
        end;
end.

```



```

        writeln('Tong cac so da nhap la: ',tong);
    end else
        writeln('Chua co so nao duoc nhap!!!');
    readln;
end.

```

Bài 36: Viết chương trình tính tổng của n số nguyên dương đầu tiên.

Giải:

```

uses crt;
var n,i,t:integer;
begin
    clrscr;
    write('n = '); readln(n); t:=0;
    for i:=1 to n do t:=t+i;
    writeln('Tong la: ',t);
    readln;
end.

```

Bài 37: Viết chương trình tính n!

Giải:

```

{$N+}
progam Giaithua;
uses crt;
var n,i:integer; gt:extended;
begin
    clrscr;
    write('n = '); readln(n); gt:=1;
    for i:=1 to n do gt:=gt*i;
    writeln(n,'! = ',gt);
    readln;
end.

```

**Chú ý:** khi muốn tính giai thừa của một số lớn (>10) thì nên sử dụng các kiểu dữ liệu có kích thước lớn (như longint, extended), trong Free Pascal có thể sử dụng kiểu int64 (là kiểu số nguyên lớn nhất) và khai báo thêm chỉ thị {\$N+} vào đầu chương trình để định hướng biên dịch nhằm cho chương trình phát sinh mã để sử dụng bộ đồng xử lý số 8087 cho việc tính toán các số thực có kích thước lớn (có thể không cần khai báo thêm chỉ thị :D ).

Bài 38: Viết chương trình nhập vào số nguyên dương n, tính:  $S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$

Bài 39: Viết chương trình nhập vào số nguyên dương n và là số chẵn, in ra giá trị của tổng sau:  $S = \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{n}$

Bài 40: Viết chương trình nhập vào các số nguyên dương n và k, in ra giá trị của tổng sau:  $S = 1 + 2^k + 3^k + \dots + n^k$ .

Bài 41: Viết chương trình nhập vào số nguyên dương n, in ra giá trị của tổng sau:  $S = 1 + 3^2 + 5^2 + \dots + (2n + 1)^2$ .

Bài 42: Viết chương trình nhập vào số nguyên dương n, in ra giá trị của biểu thức sau:  $S = 2^2 - 4^2 + 6^2 - 8^2 + \dots + (-1)^{n-1}(2n)^2$ .



Bài 43: Dùng vòng lặp while viết chương trình in ra một câu hỏi (troll nhau chẳng hạn ☺) và chương trình chỉ dừng lại khi người dùng trả lời 'y' (Yes).

Giải:

```
uses crt;
var c:char;
begin
    clrscr;
    c:='n';
    while (c<>'y') and (c<>'Y') do
        begin
            writeln('Do you love me??? (y,n): ');
            readln(c);
        end;
    readln;
end.
```

Bài 44\* (đọc thêm): Cho biết khai triển Maclaurin của hàm số sơ cấp  $\ln(x)$  như sau:

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \dots + (-1)^{n-1} \frac{x^n}{n} + (-1)^n \frac{x^{n+1}}{(n+1)(1+\theta x)^{n+1}} \quad (0 < \theta < 1)$$

Dùng vòng lặp while viết chương trình tính  $\ln(1,5)$  với độ chính xác  $\varepsilon = 10^{-5}$ .

Giải:

Dư số:  $|R_n(x)| = \left| (-1)^n \frac{x^{n+1}}{(n+1)(1+\theta x)^{n+1}} \right| \leq \left| \frac{x^{n+1}}{n+1} \right|$ . Cho  $x = 0,5$  để tính  $\ln(1,5)$  với độ chính xác đến  $10^{-5}$  ta phải chọn  $n$  sao cho:  $\left| \frac{(0,5)^{n+1}}{n+1} \right| \leq 10^{-5}$ . Khi đó:  $R_n(0,5) \leq \left| \frac{(0,5)^{n+1}}{n+1} \right| \leq 10^{-5}$ .

Chương trình:

```
uses crt;
const epsi=0,00001;
var x,r,s:real; d,n:integer;
begin
    write('Nhập x = '); readln(x);
    s:=x; n:=1; r:=sqr(0.5)/2; d:=1;
    while r>epsi do
        begin
            d:=-d; inc(n);
            s:=s+d*exp(n*ln(n))/n;
            r:=exp((n+1)*ln(x))/(n+1);
        end;
    writeln('Khai trien den cap ',n:');
    writeln('ln(',1+x:6:2,') = ',s:8:6);
    readln;
end.
```

Bài 45\*: Cho biết khai triển Maclaurin của các hàm số sơ cấp như sau:

$$e^x = 1 + x + \frac{x^2}{2!} + \dots + \frac{x^n}{n!} + \frac{e^{\theta x}}{(n+1)!} x^{n+1} \quad (0 < \theta < 1)$$

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^{k-1} \frac{x^{2k-1}}{(2k-1)!} + \frac{\sin(\theta x + (2k+1)\frac{\pi}{2})}{(2k+1)!} x^{2k+1} \quad (0 < \theta < 1)$$



$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots + (-1)^k \frac{x^{2k}}{(2k)!} + \frac{\cos(\theta x + (k+1)\pi)}{(2k+2)!} x^{2k+2} \quad (0 < \theta < 1)$$

Dùng vòng lặp while viết chương trình:

- Tính số e với độ chính xác  $\varepsilon = 10^{-5}$ .
- Tính số  $e^x$  với độ chính xác  $\varepsilon = 10^{-5}$ .
- Tính  $\sin 45^\circ$  với độ chính xác  $\varepsilon = 10^{-5}$ .
- Tính  $\cos 10^\circ$  với độ chính xác  $\varepsilon = 10^{-5}$ .

Bài 46: Viết chương trình tạo ra máy tính con có bốn phép tính: cộng, trừ, nhân và chia.

Giải:

```

program bt;
uses crt;
var n:integer; x,y,z:real;
begin
    clrscr;
    writeln('Chương trình máy tính con...');
    write('Nhập hai số thực: '); readln(x,y);
    writeln('1. Phép cộng', '2. Phép trừ', '3. Phép nhân', '4. Phép chia');
    writeln('0. Chấm dứt chương trình');
    writeln('Chọn phép toán (1 - 4): ');
    readln(n);
    case n of
        1: begin
            z:=x+y;
            writeln(x:6:2,'+',y:6:2,'=',z:6:2);
        end;
        2: begin
            z:=x-y;
            writeln(x:6:2,'-',y:6:2,'=',z:6:2);
        end;
        3: begin
            z:=x*y;
            writeln(x:6:2,'*',y:6:2,'=',z:6:2);
        end;
        4: if y=0 then writeln('Mau số bằng 0, không chia được') else
            begin
                z:=x/y;
                writeln(x:6:2,'/',y:6:2,'=',z:6:2);
            end;
        0: writeln('Chào tạm biệt...');
        else writeln('Bạn đã nhập sai số!');
    end;
    readln;
end.
    
```

Bài 47: Viết chương trình tạo máy tính con có các phép tính trên số nguyên theo menu sau:

- Lấy phần nguyên của phép chia số x cho số y: x DIV y.



b) Lấy phần dư của phép chia số  $x$  cho số  $y$ :  $x \text{ MOD } y$ .

c) Kiểm tra số nhập là lẻ hay chẵn:  $\text{ODD}(x)$ .

d) Lấy phần tử đứng sau:  $\text{SUCC}(x)$ .

e) Lấy phần tử đứng trước:  $\text{PRED}(x)$ .

📖 Bài 48: Viết chương trình tạo máy tính con có các phép tính trên các số thực theo menu sau:

a) Lấy giá trị tuyệt đối của một số thực:  $\text{ABS}(x)$ .

b) Bình phương của một số thực:  $\text{SQR}(x)$ .

c) Căn bậc hai của một số thực:  $\text{SQRT}(x)$ .

d) Làm tròn một số thực:  $\text{ROUND}(x)$ .

e) Lấy phần nguyên của một số thực:  $\text{TRUNC}(x)$ .

📖 Bài 49: Viết chương trình tạo máy tính con có các phép tính trên số thực theo menu sau:

a) Tính  $\sin x$ :  $\text{SIN}(x)$ .

b) Tính  $\cos x$ :  $\text{COS}(x)$ .

c) Tính  $\arctan x$ :  $\text{ARCTAN}(x)$ .

d) Tính  $\ln(x)$ :  $\text{LN}(x)$ .

e) Tính  $e^x$ :  $\text{EXP}(x)$ .

📖 Bài 50: Viết chương trình tính số ngày của một tháng trong năm, biết rằng:

- Tháng 4, 6, 9, 11 có 30 ngày.

- Tháng 1, 3, 5, 7, 8, 10, 12 có 31 ngày.

- Tháng 2 được tính như sau:

+ Nếu lấy số năm chia cho 4 có dư số là 0 thì tháng 2 có 29 ngày.

+ Nếu lấy số năm chia cho 4 có dư số là 1, 2, 3 thì tháng 2 có 28 ngày.

📖 Bài 51: Viết chương trình tính:

$$v = \begin{cases} x + y & \text{nếu } x > 1 \text{ và } y > 1 \\ x - y & \text{nếu } x > 1 \text{ và } y \leq 1 \\ -x + y & \text{nếu } x \leq 1 \text{ và } y > 0 \\ -x - y & \text{nếu } x \leq 1 \text{ và } y \leq 0 \end{cases}$$

📖 Bài 52: Cho cấp số cộng:  $a_0, a_1, a_2, \dots, a_n$  trong đó  $a_k = a_{k-1} + d, k = 1, 2, \dots$ . Hãy viết chương trình:

- Nhập 3 số nguyên  $a_0, d$  và  $b$ .

- Tìm và đưa ra màn hình số  $n$  nguyên không âm nhỏ nhất thoả mãn  $a_n \geq b$ .

📖 Bài 53: Lập trình nhập số thực  $x$  và số nguyên dương  $n$  từ bàn phím. Tính và đưa ra màn hình giá trị biểu thức sau:

$$P = 1 + 2x + 3x^2 + 4x^3 + \dots + nx^{n-1} + (n+1)x^n$$

### ĐỌC THÊM.

#### I/ Hàm KeyPressed:

- Dùng hàm KeyPressed của Unit Crt để trả lời câu hỏi: “Có phải vừa có một phím được nhấn từ bàn phím?”

- Cú pháp: **KeyPressed:boolean;**

- Trả về True nếu có một phím được nhấn, trả về False cho ngược lại.

- Bài tập ví dụ:

📖 Bài 54: Viết chương trình làm đầy màn hình bằng các ký tự “Xx” cho đến khi nhấn một phím từ bàn phím?

Giải:

```
uses crt;
```



```
begin
    clrscr;
    repeat write('Xx');
        until keypressed;
end.
```

## **II/ Các bài tập chữ chạy:**

Bài 55: Viết chương trình in ra giữa màn hình chữ “PASCAL” chạy từ dòng 1 đến dòng 25.

Giải:

```
uses crt;
var i:integer;
begin
    clrscr;
    textcolor(red);
    for i:=1 to 25 do
        begin
            clrscr;
            gotoxy(36,i);
            write('PASCAL');
            delay(1000);
        end;
    readln;
end.
```

Bài 56: Viết chương trình in ra màn hình chữ “PASCAL” chạy từ cột 1 đến cột 80?

Bài 57: Dùng vòng lặp while viết chương trình in ra giữa màn hình chữ “PASCAL” chạy từ dòng 1 đến dòng 25 rồi lại chạy tiếp từ dòng 1 cho đến khi nhấn một phím bất kì thì dừng lại?

Giải:

```
uses crt;
var i:integer; t:boolean;
begin
    textcolor(red); i:=1; t:=true;
    while t do
        begin
            clrscr; gotoxy(36,i);
            write('PASCAL'); delay(1000); inc(i);
            if i=25 then i:=1;
            if keypressed then t:=false;
        end;
    readln;
end.
```

Bài 58: Dùng vòng lặp while viết chương trình in ra giữa màn hình chữ “PASCAL” chạy liên tục từ cột 1 đến cột 80 cho đến khi nhấn một phím bất kì thì dừng lại?

Bài 59: Dùng vòng lặp Repeat...Until viết chương trình in ra giữa màn hình chữ “PASCAL” chạy từ dòng 1 đến dòng 25 rồi lại chạy tiếp từ dòng 1 cho đến khi nhấn một phím bất kì thì dừng lại?

Giải:

```
uses crt;
```



```

var i:integer; t:boolean;
begin
    textcolor(red);
    i:=1; t:=false;
    repeat
        clrscr; gotoxy(36,i);
        write('PASCAL'); delay(1000); inc(i);
        if i=25 then i:=1;
        if keypressed then t:=true;
    until t;
    readln;
end.

```

## CHƯƠNG IV: KIỂU DỮ LIỆU CÓ CẤU TRÚC

### BÀI 1: KIỂU MẢNG

#### I/ Kiểu mảng một chiều:

1. Định nghĩa: Mảng một chiều là dãy hữu hạn các phần tử cùng kiểu. Mảng được đặt tên và mỗi phần tử của nó có một chỉ số. Để mô tả mảng một chiều cần xác định kiểu các phần tử và cách đánh số các phần tử của nó.

#### 2. Khai báo:

- Khai báo trực tiếp: **Var <tên biến mảng>:Array[<kiểu chỉ số>] Of <kiểu phần tử>;**
- Khai báo gián tiếp: **Type <tên kiểu mảng>=Array[<kiểu chỉ số>] Of <kiểu phần tử>;**  
**Var <tên biến mảng> : <tên kiểu mảng>;**

3. Truy nhập vào một phần tử của mảng: **<tên biến mảng>[<chỉ số>] .**

#### 4. Tìm kiếm trên mảng:

a) Dãy thuật tìm kiếm tuần tự: Ta bắt đầu tìm kiếm từ phần tử đầu tiên. Nếu khoá của phần tử này không phải là khoá cần tìm thì ta đến phần tử kế tiếp và cứ như thế cho đến khi nào ta đến phần tử có khoá cần tìm (tìm kiếm thành công) hoặc cho đến khi đi hết danh sách (tìm kiếm thất bại).

b) Giải thuật tìm kiếm nhị phân: Để thực hiện dãy thuật tìm kiếm nhị phân thì mảng phải có thứ tự. Để tìm kiếm phần tử có khoá là k ta lần lượt thực hiện các bước sau:

- Bước 1: Phạm vi tìm kiếm ban đầu là toàn bộ mảng.
- Bước 2: Lấy khoá của phần tử ở chính giữa của phạm vi tìm kiếm gọi là x và đem so sánh với k: nếu x=k thì tìm thấy khoá này, giải thuật kết thúc thành công. Nếu x<k thì phạm vi tìm kiếm mới là các phần tử nằm phía trước x. Nếu x>k thì phạm vi tìm kiếm mới là các phần tử nằm phía sau x.
- Bước 3: Nếu tồn tại một phạm vi tìm kiếm mới thì lặp lại bước 2, ngược lại giải thuật kết thúc không thành công, tìm kiếm thất bại.

#### II/ Kiểu mảng hai chiều:

1. Định nghĩa: Mảng hai chiều là bảng các phần tử cùng kiểu.

Ví dụ:

	1	2	3	4	5	... (chỉ số cột)
1	5	7	9	3	4	
2	2	7	6	3	1	
3	7	9	3	8	2	





4 

1	5	7	4	9
---	---	---	---	---

  
... (chỉ số hàng)

## 2. Khai báo:

- Khai báo trực tiếp:

**Var <tên biến mảng>:Array[<kiểu chỉ số hàng>,<kiểu chỉ số cột>] Of <kiểu phần tử>;**

- Khai báo gián tiếp:

**Type <tên kiểu mảng>=Array[<kiểu chỉ số hàng>,<kiểu chỉ số cột>] Of <kiểu phần tử>;**

**Var <tên biến mảng>:<tên kiểu mảng>;**

**3. Tham chiếu đến phần tử của mảng hai chiều:** <tên biến mảng>[<chỉ số hàng>,<chỉ số cột>] .

## III/ Bộ sinh số ngẫu nhiên:

Cú pháp:

**Randomize;**

**x:=Random(<range>);**

<range> là một số nguyên kiểu Word. Giá trị trả về x là một số nguyên kiểu Word có giá trị  $0 \leq x \leq \text{range}$ .

## BÀI 2: KIỂU XÂU

**I/ Khái niệm:** Xâu là dãy các kí tự trong bộ mã ASCII, mỗi kí tự được gọi là một phần tử của xâu. Số lượng kí tự trong một xâu được gọi là độ dài của xâu, xâu có độ dài bằng 0 gọi là xâu rỗng. Có thể xem xâu là mảng một chiều mà mỗi phần tử là một kí tự. Các kí tự của xâu được đánh số thứ tự, thường bắt đầu từ 1. Độ dài lớn nhất của xâu không vượt quá 255.

**II/ Khai báo:** **Var <tên biến>:String[<độ dài lớn nhất của xâu>;**

Chú ý: Trong một xâu có thể bỏ qua phần khai báo độ dài. Khi đó độ dài lớn nhất của xâu sẽ nhận giá trị ngầm định là 255.

**III/ Tham chiếu đến phần tử của xâu:** <tên biến xâu>[<chỉ số>] .

## IV/ Các thao tác xử lí xâu:

**1. Phép ghép xâu:** Được sử dụng để ghép nhiều xâu thành một, có thể thực hiện phép ghép xâu đối với các hằng và biến xâu. Kí hiệu: +

VD: 'THPT' + 'Chuyen' + 'Hung Vuong' + ' - ' + ' Binh Duong.' = 'THPT Chuyen Hung Vuong – Binh Duong.'

**2. Các phép so sánh (=, <>, <, >, <=, >=):** Có thứ tự ưu tiên phép tính thấp hơn phép ghép xâu. Quy tắc: xâu A là lớn hơn xâu B nếu kí tự đầu tiên khác nhau giữa chúng kể từ trái sang trong xâu A có mã ASCII lớn hơn. Nếu A và B là các xâu có độ dài khác nhau và A là đoạn đầu của B thì A nhỏ hơn B. Hai xâu được coi là bằng nhau nếu như chúng giống nhau hoàn toàn.

## V/ Thủ tục và hàm chuẩn để xử lí xâu:

- **Delete(<st>,<vt>,<n>):** thực hiện việc xóa n kí tự của biến xâu st bắt đầu từ vị trí vt.
- **Insert(<s1>,<s2>,<vt>):** chèn xâu s1 vào xâu s2, bắt đầu từ vị trí vt.
- **Copy(<s>,<vt>,<n>):** tạo xâu gồm n kí tự liên tiếp bắt đầu từ vị trí vt của xâu s.
- **Length(<s>):** cho giá trị là độ dài xâu s.
- **Pos(<s1>,<s2>):** cho vị trí xuất hiện đầu tiên của xâu s1 trong xâu s2.
- **Uppcase(<ch>):** cho chữ cái in hoa tương ứng với chữ cái trong ch.

## BÀI 3: KIỂU BẢN GHI

**I/ Khái niệm:** Dữ liệu bản ghi dùng để miêu tả các đối tượng có cùng một số thuộc tính mà các thuộc tính có thể có các kiểu dữ liệu khác nhau. Mỗi thuộc tính của đối tượng tương ứng với một trường của bản ghi. Các trường khác nhau có thể có các kiểu dữ liệu khác nhau. Mỗi trường được tham khảo thông qua tên của nó (thay vì thông qua chỉ số như của mảng).

**II/ Khai báo:**

Type <tên kiểu bản ghi>=Record  
     <tên trường 1> : <kiểu trường 1>;  
     <tên trường 2> : <kiểu trường 2>;  
     [...]  
     <tên trường n> : <kiểu trường n>;

End;

Var <tên biến bản ghi> : <tên kiểu bản ghi>;

**III/ Truy cập vào một phần tử của bản ghi:**

1. Cú pháp:                      <tên biến>.<tên trường>;

2. Chú ý: các phép tính thực hiện được trên mỗi trường là các phép tính thực hiện được trên kiểu dữ liệu của trường đó.

**IV/ Câu lệnh With:** Cho phép truy nhập các trường của bản ghi một cách vắn tắt hơn.

With <tên biến bản ghi> Do <câu lệnh>;

Chú ý: Việc sử dụng câu lệnh With tuy làm chương trình ngắn gọn, vắn tắt hơn, nhưng dễ gây nhầm lẫn, sai sót khi có biến được khai báo cùng với tên trường của bản ghi.

**V/ Thao tác trên toàn bản ghi:**

- Có thể dùng phép gán để sao chép giá trị của các trường trong bản ghi này sang các trường tương ứng trong bản ghi khác như sau:   <biến bản ghi 2>:=<biến bản ghi 1>;

- Chú ý: không thể dùng Write, Writeln, Read, Readln để xuất/nhập cho một biến bản ghi mà chỉ có thể dùng cho việc xuất/nhập trên các trường của bản ghi.

*BÀI TẬP*

Bài 1: Viết chương trình tính và đưa ra màn hình bảng nhân?

Giải:

```
uses crt;
var b:array[1..9,1..10] of integer; i,j:integer;
begin
    clrscr;
    for i:=1 to 9 do
        for j:=1 to 10 do b[i,j]:=i*j;
    for i:=1 to 9 do
        begin
            for j:=1 to 10 do write(b[i,j]:4);
            writeln;
        end;
    readln;
end.
```

Bài 2: Viết chương trình tìm phần tử có giá trị lớn nhất trong mảng và đưa ra màn hình chỉ số và giá trị của phần tử tìm được. Nếu có nhiều phần tử có cùng giá trị lớn nhất thì đưa ra phần tử có giá trị nhỏ nhất?

Giải:

```
uses crt;
const Nmax=100;
type ArrInt=array[1..Nmax] of integer;
var a:ArrInt; n,i,j:integer;
```



```

begin
    clrscr;
    write('Nhap so luong phan tu cua day so, N = '); readln(n);
    for i:=1 to n do
        begin
            write('Phan tu thu ',i,' = '); readln(a[i]);
        end;
    j:=1;
    for i:=2 to n do
        if a[i]>a[j] then j:=i;
    writeln('Chi so max: ',j);
    writeln('Gia tri max: ',a[j]);
    readln;
end.

```

Bài 3: Tạo mảng ngẫu nhiên gồm n số nguyên rồi sắp xếp dãy số đó?

Giải:

```

uses crt;
var a:array[1..250] of integer; n,i,j,t:integer;
begin
    clrscr;
    randomize;
    write('Nhap N = '); readln(n);
    for i:=1 to n do a[i]:=random(300)-random(300);
    for i:=1 to n do write(a[i]:5);
    writeln;
    for j:=n downto 2 do
        for i:=1 to j-1 do
            if a[i]>a[i+1] then
                begin
                    t:=a[i];
                    a[i]:=a[i+1];
                    a[i+1]:=t;
                end;
        writeln('Day so da duoc sap xep:');
    for i:=1 to n do write(a[i]:5);
    writeln;
    readln;
end.

```

end.

Bài 4: Cho mảng A gồm n phần tử. Hãy viết chương trình tạo mảng B, trong đó B[i] là tổng của i phần tử đầu tiên của A?

Giải:

```

uses crt;
var a,b:array[1..250] of integer; n,i:integer;
begin
    clrscr;

```



```

write('Cho biet so luong phan tu cua A, N = '); readln(n);
for i:=1 to n do
begin
    write('A['i,'] = '); readln(a[i]);
end;
b[1]:=a[1];
for i:=2 to n do b[i]:=b[i-1] + a[i];
for i:=1 to n do write(b[i]:6);
readln;
end.

```

Bài 5: Viết chương trình nhập họ tên của hai người vào hai biến xâu và đưa ra màn hình xâu dài hơn, nếu bằng nhau thì đưa ra xâu nhập sau?

Giải:

```

uses crt;
var a,b:string;
begin
    clrscr;
    write('Nhap ho ten nguoi thu nhat: '); readln(a);
    write('Nhap ho ten nguoi thu hai: '); readln(b);
    if length(a) > length(b) then writeln(a) else writeln(b);
    readln;
end.

```

Bài 6: Viết chương trình nhập vào hai xâu từ bàn phím và kiểm tra kí tự đầu tiên của xâu thứ nhất có trùng với kí tự cuối cùng của xâu thứ hai không?

Giải:

```

uses crt;
var a,b:string;
begin
    clrscr;
    write('Nhap xau thu nhat: '); readln(a);
    write('Nhap xau thu hai: '); readln(b);
    if a[1]=b[length(b)] then writeln('Trung nhau') else writeln('Khac nhau');
    readln;
end.

```

Bài 7: Viết chương trình nhập vào một xâu từ bàn phím và đưa ra màn hình xâu đó nhưng được viết theo thứ tự ngược lại?

Giải:

```

uses crt;
var a:string; i:integer;
begin
    clrscr;
    write('Nhap xau: '); readln(a);
    for i:=length(a) downto 1 do write(a[i]);
    readln;
end.

```



Bài 8: Viết chương trình nhập vào từ bàn phím một xâu. Kiểm tra xâu đó có phải là xâu đối xứng hay không? Xâu đối xứng có tính chất: đọc nó từ trái sang phải cũng thu được kết quả giống như đọc từ phải sang trái (còn gọi là xâu palindrome).

Giải:

```
uses crt;
var a,p:string; i:integer;
begin
    clrscr;
    write('Nhap vao xau: '); readln(a);
    p:= '';
    for i:=length(a) downto 1 do p:=p+a[i];
    if a=p then writeln('Xau la palindrome') else
        writeln('Xau khong la palindrome');
    readln;
end.
```

Bài 9: Nhập vào từ bàn phím một xâu, thay thế tất cả các cụm kí tự “anh” bằng cụm kí tự “em”? (sử dụng hàm pos ☺)

Bài 10: Viết chương trình nhập vào từ bàn phím thông tin của từng học sinh trong lớp, thực hiện xếp loại và đưa ra màn hình kết quả xếp loại học sinh?

Giải:

```
uses crt;
type hs=record
    hoten:string[30];
    ngaysinh:string[10];
    diachi:string[50];
    toan,van:real;
    xl:char;
end;
var lop:array[1..60] of hs; n,i:integer;
begin
    clrscr;
    write('Nhap so luong hoc sinh cua lop, N = '); readln(n);
    for i:=1 to n do
        begin
            writeln('Nhap du lieu ve hoc sinh thu ',i);
            write('Ho ten: '); readln(lop[i].hoten);
            write('Ngay sinh: '); readln(lop[i].ngaysinh);
            write('Dia chi: '); readln(lop[i].diachi);
            write('Diem toan: '); readln(lop[i].toan);
            write('Diem van: '); readln(lop[i].van);
            if lop[i].toan+lop[i].van >= 18 then lop[i].xl:='A' else
                if lop[i].toan+lop[i].van >= 14 then lop[i].xl:='B' else
                    if lop[i].toan+lop[i].van >= 10 then lop[i].xl:='C' else
                        lop[i].xl:='D';
        end;
    end;
```



```

clrscr;
writeln('Danh sach xep loai hoc sinh trong lop: ');
for i:=1 to n do writeln(lop[i].hoten:30,' – Xep loai: ',lop[i].xl);
readln;

```

end.

Bài 11: Viết chương trình nhập vào mảng A gồm n phần tử ( $n < 10$ ) số nguyên có giá trị tuyệt đối không lớn hơn 100. Hãy cho biết các phần tử của mảng này có lập thành một cấp số cộng hoặc cấp số nhân hay không?

Giải:

```

uses crt;
var a:array[1..10] of integer; n,i:integer;
begin
    clrscr;
    write('Nhap so luong phan tu cua mang: '); readln(n);
    for i:=1 to n do
        begin
            write('Phan tu thu ',i,' la: '); readln(a[i]);
        end;
    for i:=2 to n do
        begin
            if (a[i] = (a[i-1] + a[i+1])/2) then writeln('Tao thanh CSC') else
                writeln('Khong tao thanh CSC');
            if (sqr(a[i]) = a[i-1].a[i+1]) then writeln('Tao thanh CSN') else
                writeln('Khong tao thanh CSN');
        end;
    readln;
end.

```

Bài 12: Dãy F gọi là dãy Fibonacci nếu: 
$$\begin{cases} F_0 = 0 \\ F_1 = 1 \\ F_n = F_{n-1} + F_{n-2} \text{ với } n \geq 2 \end{cases}$$
. Viết chương trình nhập từ bàn phím số nguyên dương N và đưa ra màn hình số hạng thứ N của dãy Fibonacci?

Giải:

```

uses crt;
var a:array[0..100] of integer; n,i:integer;
begin
    clrscr;
    write('Nhap N = '); readln(n);
    a[0]:=0; a[1]:=1;
    for i:=2 to n do a[i]:=a[i-1] + a[i-2];
    writeln('So hang thu ',n,' la: ',a[n]);
    readln;
end.

```

Bài 13: Viết chương trình thực hiện công việc sau đây: Nhập vào một mảng các số nguyên, sau đó đưa ra màn hình:

a) Giá trị trung bình của dãy.



b) Giá trị của phần tử dương đầu tiên và chỉ số của nó?

Giải:

```
uses crt;
var a:array[1..100] of integer; n,i:integer; tong:longint; tb:real;
begin
    clrscr;
    writeln('So luong phan tu trong mang: '); readln(n);
    for i:=1 to n do
        begin
            write('Phan tu thu ',i,' la: '); readln(a[i]);
        end;
    for i:=1 to n do tong:=tong+a[i];
    tb:=tong/n;
    writeln('Gia tri trung binh cua day: ',tb:10:2);
    i:=1;
    while (i<=n) and (a[i]<=0) do inc(i);
    if a[i]>0 then writeln('Phan tu duong dau tien la: ',a[i], ' o vi tri thu ',i) else
        writeln('Mang ban vua nhap khong co so duong');
    readln;
end.
```

Bài 14: Lập trình nhập vào một mảng gồm N phần tử ( $n < 15$ ), mỗi phần tử là một số nguyên nhỏ hơn 100. Đếm xem có bao nhiêu phần tử nhỏ hơn k, bằng k, với k là một số nguyên dương cho trước?

Giải:

```
uses crt;
var a:array[1..15] of integer; n,i,k,dem:integer;
begin
    clrscr;
    write('N = '); readln(n);
    for i:=1 to n do
        begin
            write('Phan tu thu ',i,' la: '); readln(a[i]);
        end;
    write('K = '); readln(k);
    dem:=0;
    for i:=1 to n do
        if a[i]<k then inc(dem);
    writeln('So phan tu nho hon k la: ',dem);
    dem:=0;
    for i:=1 to n do
        if a[i]=k then inc(dem);
    writeln('So phan tu bang k la: ',dem);
    readln;
end.
```



Bài 15: Viết chương trình thực hiện công việc sau đây: Nhập vào một mảng các số nguyên sau đó đưa ra màn hình:

a) Phần tử lớn nhất, phần tử nhỏ nhất của mảng.

b) In ra các cặp phần tử liên tiếp có trị tuyệt đối của hiệu nhỏ hơn k, với  $k > 0$  cho trước.

Hướng dẫn:

a) `max:=a[1]; min:=a[1];`

`for i:=1 to n do`

`begin`

`if a[i] > max then max:=a[i];`

`if a[i] < min then min:=a[i];`

`end;`

b) `for i:=1 to n do`

`if (abs(a[i]-a[i+1]) < k) then writeln(a[i], ' ', a[i+1]);`

Bài 16: Nhập vào một mảng các số nguyên, sau đó đưa ra màn hình:

a) Cho biết mảng đó có phải là mảng tăng hay mảng giảm không? Biết:

b) Mảng A gọi là tăng nếu ta có  $A[i] \leq A[i+1]$  với  $i \in [1, n-1]$ .

c) Mảng A gọi là mảng giảm nếu  $A[i] > A[i+1]$  với  $i \in [1, n-1]$ .

d) Cho biết mảng đó có phải là mảng đan dấu hay không? Biết: mảng A gọi là đan dấu nếu tích của hai phần tử liên tiếp là một số âm.

Hướng dẫn:

`{kiem tra mang tang}`

`i:=2;`

`while (i <= n) and (a[i-1] <= a[i]) do inc(i);`

`if i > n then writeln('Mang tang') else writeln('Khong la mang tang');`

`{kiem tra mang giam}`

`i:=2;`

`while (i <= n) and (a[i-1] > a[i]) do inc(i);`

`if i > n then writeln('Mang giam') else writeln('Khong la mang giam');`

`{kiem tra mang dan dau}`

`i:=2;`

`while (i <= n) and (a[i-1].a[i] < 0) do inc(i);`

`if i > n then writeln('Mang dan dau') else writeln('Khong la mang dan dau');`

Bài 17: Viết chương trình nhập vào một mảng các số nguyên sau đó loại bỏ những phần tử có giá trị giống nhau. Đưa ra màn hình kết quả?

Giải:

`uses crt;`

`var a:array[1..100] of integer; i,j,n:integer; ok:boolean;`

`begin`

`clrscr;`

`write('N = '); readln(n);`

`for i:=1 to n do`

`begin`

`write('Phan tu thu ', i, ' : '); readln(a[i]);`

`end;`

`for i:=1 to n do`





```

begin
    ok:=true;
    for j:=1 to i-1 do
        if a[i]=a[j] then ok:=false;
    if ok then write(a[i]:4);
end;
readln;
end.

```

Bài 18: Viết chương trình nhập vào một xâu và in ra màn hình xâu đó bằng các kí tự in hoa?

Giải:

```

uses crt;
var s1,s2:string; i,dai:integer;
begin
    clrscr;
    write('Nhap vao mot xau: '); readln(s1);
    dai:=length(s1); s2:='';
    for i:=1 to dai do s2:=s2+upcase(s1[i]);
    writeln('Xau viet hoa: ',s2);
    readln;
end.

```

Bài 19: Viết chương trình nhập vào một xâu và in ra màn hình xâu kết quả có các kí tự đầu là chữ hoa, các kí tự sau là chữ thường?

Hướng dẫn:

```

for i:=1 to length(s) do
    if s[i]<>' ' then
        if i=1 then s[i]:=upcase(s[i]) else
            if s[i-1]=' ' then s[i]:=upcase(s[i]) else
                if s[i]=upcase(s[i]) then s[i]:=chr(ord(s[i])+32));
writeln('Xau moi la: ',s);

```

Bài 20: Viết chương trình “nắn xâu” theo nghĩa sau:

- Nhập vào một xâu.
- Bỏ các kí tự trống bên trái xâu.
- Bỏ các kí tự trống bên phải xâu.
- Trong xâu, nếu có hơn một dấu cách liên tiếp thì xoá bớt, chỉ để lại một dấu cách.

Hướng dẫn:

```

if length(s)>0 then
begin
    while s[1]=' ' do delete(s,1,1);
    while s[length(s)]=' ' do delete(s,length(s),1);
end;
while pos(' ',s)>0 do delete(s,pos(' ',s),1);

```

Bài 21: Viết chương trình nhập vào hai từ và kiểm tra xem chúng có phải là annagram của nhau hay không? (annagram nghĩa là có thể tạo từ này bằng cách hoán vị các kí tự của từ kia. Ví dụ: dear, read, dare).

Hướng dẫn:



```
dai:=length(s1); i:=1; ok:=true;
while (i<=dai) and ok do
  begin
    j:=i;
    while (j<=dai) and (s1[i]<>s2[j]) do inc(j);
    if j>dai then ok:=false else
      begin
        tmp:=s2[j]; s2[j]:=s2[i]; s2[i]:=tmp;
      end;
    inc(i);
  end;
if ok then writeln('La annagram') else writeln('Khong la annagram');
```

Bài 22: Viết chương trình nhập vào một xâu rồi đưa ra màn hình số kí tự khác nhau trong xâu đó?

Giải:

```
uses crt;
var s:string; i,j,d:integer;
begin
  clrscr;
  write('Nhap vao mot xau: '); readln(s);
  if length(s)=0 then writeln('Xau rong') else
    begin
      d:=1;
      for i:=2 to length(s) do
        begin
          j:=i-1;
          while (j>0) and (s[j]<>s[i]) do j:=j-1;
          if j=0 then inc(d);
        end;
      writeln('Xau vua nhap co ',d,' ki tu khac nhau.');    end;
  readln;
end.
```

Bài 23: Viết chương trình nhập vào một mảng số nguyên ( $n < 100$ )

- In ra theo thứ tự ngược lại?
- Tính tổng của các phần tử của mảng?
- Liệt kê các phần tử âm, dương?

Bài 24: Viết chương trình nhập vào một mảng số nguyên ( $n < 100$ )

- Tìm số lớn nhất, số nhỏ nhất trong mảng?
- Tìm số âm đầu tiên của mảng?

Bài 25: Để giữ bí mật người ta phải mã hoá các thông tin trước khi truyền đi hoặc lưu trữ. Một trong những cách mã hoá sớm nhất được sử dụng rộng rãi thời cổ đại là cách mã hoá do Xê Da đề xuất: Trong thông điệp, người ta thay đổi mỗi chữ cái bằng chữ cái đứng sau nó k vị trí trong bảng chữ cái. Việc tìm kiếm thay thế được tiến hành vòng tròn theo bảng chữ cái. Nếu bảng chữ cái có



N chữ, thì sau chữ cái thứ N-1 là chữ cái N, sau chữ cái N là chữ cái thứ nhất,... Cách mã hoá này được gọi là mã Xê Da. Các kí tự ngoài bảng chữ cái vẫn được giữ nguyên. Hãy lập trình:

Nhập từ bàn phím số nguyên k ( $1 < k \leq 26$ ) và xâu s không quá 255 kí tự.

Mã hoá theo qui tắc mã Xê Da và đưa ra màn hình.

Giải:

```
uses crt;
const p='abcdefghijklmnopqrstuvwxyz';
var s:string; i,j,k:integer;
begin
    clrscr;
    write('Nhap vao xau: '); readln(s);
    write('Nhap vao k: '); readln(k);
    for i:=1 to n do
        begin
            j:=pos(s[i],p)+k;
            s[i]:=p[j];
        end;
    writeln('Xau da ma hoa: ',s);
    readln;
end.
```

## CHƯƠNG V: TỆP VÀ THAO TÁC VỚI TỆP

### I/ Kiểu dữ liệu tệp:

#### 1. Vai trò:

- Tất cả các dữ liệu thuộc các kiểu dữ liệu đã xét đều được lưu trữ trong bộ nhớ trong (RAM – Random Access Memory), dữ liệu sẽ bị mất khi tắt máy.
- Với những bài toán có khối lượng dữ liệu lớn, có yêu cầu lưu trữ để xử lí nhiều lần, cần có kiểu dữ liệu tệp (file).
- Đặc điểm của file:
  - + Được lưu trữ lâu dài ở bộ nhớ ngoài (đĩa từ, CD – Compact Disk, băng từ,...) và không bị mất khi tắt nguồn điện.
  - + Lượng dữ liệu lưu trữ có thể rất lớn tùy thuộc vào dung lượng đĩa.

#### 2. Phân loại tệp và thao tác với tệp:

- Tệp là một cấu trúc dữ liệu gồm nhiều phần tử cùng kiểu được chứa trên thiết bị nhớ ngoài.
- Số lượng phần tử không hạn chế.
- Mỗi phần tử của tệp là một dữ liệu thuộc kiểu cơ sở.

#### — Xét theo tổ chức dữ liệu, có thể phân tệp thành hai loại:

- Tệp văn bản: dữ liệu được ghi dưới dạng các kí tự theo mã ASCII.
- Tệp có cấu trúc: thành phần của tệp được tổ chức theo một cấu trúc nhất định. Tệp nhị phân là một trường hợp riêng của tệp có cấu trúc.

#### — Xét theo cách truy cập, có thể phân tệp thành hai loại:

- Tệp truy cập tuần tự: truy cập đến một dữ liệu nào đó trong tệp chỉ bằng cách bắt đầu từ tệp và đi qua lần lượt các dữ liệu trước nó.



- Tệp truy cập trực tiếp: tham chiếu đến dữ liệu cần truy cập bằng cách xác định vị trí trực tiếp của những dữ liệu đó.
- Số lượng phần tử của tệp không cần xác định trước.
- Hai thao tác cơ bản đối với tệp là **đọc/ghi dữ liệu tệp**. **Quy trình:** Khai báo biến tệp => Mở tệp => Đọc/ghi dữ liệu => Đóng tệp.

## II/ Thao tác với tệp:

1. Khai báo: **Var <tên biến tệp>:Text;**

2. Thao tác với tệp:

a) Gắn tên tệp: **Assign(<biến tệp>,<tên tệp>);**

- Tên tệp là biến xâu hoặc hằng xâu.
- Ta không thao tác trực tiếp với dữ liệu trên đĩa mà thông qua biến tệp. Biến tệp là một biến kiểu tệp.
- Tên tệp là một biểu thức kiểu String mô tả tên tệp (bao gồm cả đường dẫn vật lý).

b) Mở tệp:

- Mở tệp để ghi dữ liệu: **Rewrite(<biến tệp>);**
- Mở tệp để đọc dữ liệu: **Reset(<biến tệp>);**

c) Đọc/ghi tệp văn bản:

- Việc đọc tệp văn bản được thực hiện giống như nhập từ bàn phím.
- Việc ghi dữ liệu ra tệp văn bản giống như ghi dữ liệu ra màn hình.
- Đọc dữ liệu:

**Read(<biến tệp>,<danh sách biến>);**

**Readln(<biến tệp>,<danh sách biến>);**

- Ghi dữ liệu:

**Write(<biến tệp>,<danh sách kết quả>);**

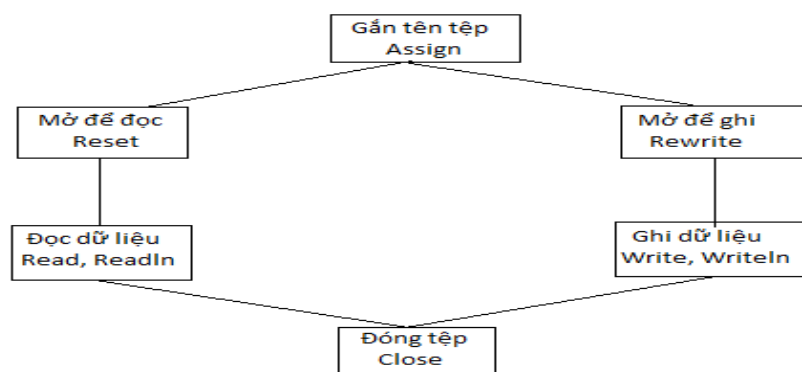
**Writeln(<biến tệp>,<danh sách kết quả>);**

d) Đóng tệp:

**Close(<biến tệp>);**

- Sau khi đóng, một tệp vẫn có thể được mở lại.
- Khi mở lại tệp, nếu vẫn dùng biến tệp cũ thì không cần phải dùng thủ tục Assign gắn lại tên tệp.

## III/ Bảng tóm tắt các thao tác với tệp:



## IV/ Một số hàm chuẩn:

- Eof(<biến tệp>):** trả về 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.
- Seek(<biến tệp>,<n>):** thực hiện việc chuyển con trỏ tệp đến vị trí thứ n.



- **Erase(<biến tệp>):** thực hiện xoá một tệp trên đĩa với biến tệp đại diện, không thể xoá một tệp đang mở.
- **Rename(<biến tệp>,<tên mới>):** thực hiện việc đổi tên tệp mà biến tệp đại diện, không thể đổi tên một tệp đang mở.
- **FilePos(<biến tệp>):** trả về vị trí con trỏ của tệp đại diện bởi biến tệp.
- **FileSize(<biến tệp>):** trả về kích thước tệp (là một số nguyên) của tệp đại diện bởi biến tệp.
- **Append(<biến tệp>):** mở để ghi nối thêm vào cuối tệp.

### *ĐỌC THÊM.* Thủ tục kiểm tra lỗi nhập/xuất:

Sử dụng các chỉ thị riêng mở rộng khả năng ngôn ngữ của Pascal {\$I+} và {\$I-} để thực hiện việc cho phép hay không cho phép chương trình dịch tự phát các mã kiểm tra kết quả của một lời gọi đến thủ tục nhập/xuất thông qua một thủ tục kiểm tra lỗi nhập/xuất qua hàm IO Result như sau:

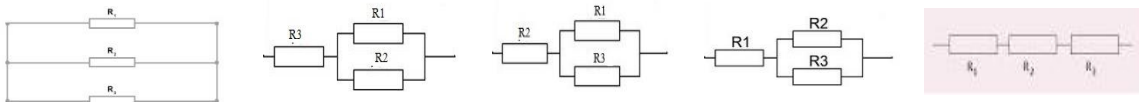
```

Procedure Kiemtra(<tên file>:String; Var loi:Boolean);
Var f:File of Byte;
Begin
    Loi:=False;
    Assign(f,<tên file>);
    {$I-}
    Reset(f);
    {$I+}
    If (IOResult <> 0) then
        Begin
            Loi:=True;
            Writeln('File nay khong ton tai');
        End Else Close(f);
End;
```

### *BÀI TẬP*

Bài 1: Cho 3 điện trở R1, R2, R3. Sử dụng ba điện trở để tạo thành 5 mạch điện có điện trở tương đương khác nhau. Cho tệp văn bản RESIST.DAT gồm nhiều dòng, mỗi dòng chứa ba số thực R1, R2, R3, các số cách nhau một dấu cách,  $0 < R1, R2, R3 < 10^5$ . Viết chương trình tính các điện trở tương đương và ghi kết quả ra tệp văn bản RESIST.EQU, mỗi dòng ghi năm điện trở tương đương của ba điện trở ở dòng dữ liệu vào tương ứng.

Các mạch điện trở:



Giải:

```

var a:array[1..5] of real; r1,r2,r3:real; i:integer; f1,f2:text;
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.

```

Bài 2: Viết chương trình đọc dữ liệu từ một file văn bản này sang một file văn bản khác.

Giải:

```

var f1,f2:text; ktke:char;
begin
    assign(f1,'INPUT.TXT'); reset(f1);
    assign(f2,'OUTPUT.TXT'); rewrite(f2);
    while not eof(f1) do
        begin
            if eoln(f1) then
                begin
                    readln(f1); writeln(f2);
                end else
                    begin
                        read(f1,ktke);
                        write(f2,ktke);
                    end;
            end;
        end;
    close(f1); close(f2);
end.

```

Bài 3: Viết chương trình thực hiện tìm số max và cho biết có bao nhiêu số max, dữ liệu lấy từ tệp input.dat.

Giải:

```

uses crt;
var f:text; n,max,count:longint;
begin
    assign(f,'INPUT.DAT'); reset(f);
    max:=-2127483647;
    while not eof(f) do
        begin
            read(f,n);
            if n>max then
                begin
                    max:=n;

```



```

        count:=1;
    end else
        if n=max then inc(count);
    end;
    close(f); clrscr;
    writeln(max,' ',count);
    readln;
end.

```

Bài 4: Viết chương trình đếm có bao nhiêu dòng trong tệp input.dat?

Giải:

```

uses crt;
var f:text; dem:integer;
begin
    clrscr;
    assign(f,'INPUT.DAT'); reset(f); dem:=0;
    while not eof(f) do
        begin
            readln(f);
            inc(dem);
        end;
    close(f);
    writeln('Tep chua ',dem,' dong. ');
    readln;
end.

```

Bài 5: Viết chương trình ghi tất cả các số nguyên tố từ 1 đến n (n nhập từ màn hình) vào tệp NGUYENTO.DAT. Yêu cầu các số nguyên tố phải được ghi vào tệp theo thứ tự từ nhỏ đến lớn, mỗi số được ghi trên một dòng và cuối chương trình xuất ra trên màn hình có bao nhiêu số nguyên tố được ghi vào tệp.

Giải:

```

uses crt;
var f:text; n,i,lim:longint; dem:integer;
begin
    clrscr;
    write('Nhap vao n: '); readln(lim);
    assign(f,'NGUYENTO.DAT'); rewrite(f); dem:=0;
    writeln('Dang liet ke cac SNT vao tep NGUYENTO.DAT.....');
    for n:=2 to lim do
        begin
            i:=2;
            while (i<=trunc(sqrt(n))) and (n mod i<>0) do inc(i);
            if (i>trunc(sqrt(n))) then
                begin
                    writeln(f,n);
                    inc(dem);
                end;
        end;
    end;
end.

```



```

end;
close(f);
writeln;
writeln('So luong SNT duoc liet ke vao tep: ',dem);
writeln('Hay mo tep NGUYENTO.DAT de kiem tra lai!');
readln;

```

end.

Bài 6: Kết quả học tập đầy đủ của các học sinh trong một lớp được lưu dưới dạng một tệp văn bản có tên là HOSO.TXT. Các thông tin của mỗi học sinh được lưu trên một dòng theo khuôn dạng sau: Họ và tên học sinh, ngày tháng năm sinh, điểm trung bình, xếp loại.

Hãy viết chương trình thực hiện các công việc:

- Đọc dữ liệu từ tệp văn bản đã cho.
- Hiện thị lên màn hình các thông tin vừa đọc theo dạng:

<i>Họ và tên</i>	<i>Ngày sinh</i>	<i>DTB</i>	<i>XL</i>
<i>Ton Ngo Khong</i>	<i>12/12/12</i>	<i>9.9</i>	<i>Gioi</i>

Giải:

```

uses crt;
var   f:text; i,dem:integer;
      hoten:string[30]; ngsinh:string[10]; DTB:real; xl:string[10];
begin
  clrscr;
  writeln; writeln;
  writeln('Bang ket qua hoc tap ca lop':50);
  writeln;
  writeln('Ho va ten','Ngày sinh':30,'DTB':12,'XL':12);
  writeln;
  assign(f,'HOSO.TXT'); reset(f); dem:=0;
  while not eof(f) do
    begin
      inc(dem);
      readln(f,hoten);
      readln(f,ngsinh);
      readln(f,dtb);
      readln(f,xl);
      writeln(hoten,'',(30-length(hoten)),ngsinh:10,dtb:10:1,xl:11);
    end;
  close(f);
  readln;
end.

```





# CHƯƠNG VI CHƯƠNG TRÌNH CON & LẬP TRÌNH CỐ CẤU TRÚC

## BÀI 1: CHƯƠNG TRÌNH CON VÀ PHÂN LOẠI

### I/ Khái niệm:

- Khi lập trình để giải các bài toán trên máy tính có thể phân chia chương trình (gọi là chương trình chính) thành các khối (modul), mỗi khối bao gồm các lệnh giải một bài toán con nào đó. Mỗi khối lệnh sẽ được xây dựng thành một chương trình con. Sau đó chương trình chính sẽ được xây dựng từ những chương trình con này.
- 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 hoá; mở rộng khả năng ngôn ngữ;
  - + Thuận tiện cho việc phát triển, nâng cấp chương trình.

### II/ Phân loại và cấu trúc của chương trình con:

#### 1. 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. Ví dụ:  $\sin(x)$ ,  $\sqrt{x}$ ,  $\text{length}(s)$ ,...
- 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: `writeln`, `readln`, `delete`, `insert`,...

Kiểu chương trình con	Khai báo	Giá trị trả về
Thủ tục	<b>Procedure</b>	Không
Hàm	<b>Function</b> Có khai báo kiểu cho tên hàm, cuối hàm thường có một lệnh gán giá trị kết quả cho tên hàm.	Có trả về một giá trị thông qua tên hàm

#### Chú ý:

- Kiểu cho tên hàm chỉ được cho phép là các kiểu đơn giản đã được Pascal định nghĩa.
- Nếu muốn chương trình con trả về kết quả là một kiểu phức tạp hơn, như mảng, thì nên dùng tham số hình thức biến.

#### 2. Cấu trúc chương trình con:

- Có cấu trúc tương tự chương trình, nhưng phải có tên và phần đầu dùng để khai báo tên, nếu là hàm thì phải khai báo kiểu dữ liệu cho giá trị trả về của hàm.

<phần đầu>

[<phần khai báo>]

<phần thân>

- Phần khai báo: có thể khai báo biến cho dữ liệu vào và ra, các hằng và biến dùng trong chương trình con.
- Phần thân: là dãy câu lệnh thực hiện để từ dữ liệu vào ta nhận được dữ liệu ra hay kết quả như mong muốn.
- Tham số hình thức:
  - + Các biến được khai báo cho dữ liệu vào/ra được gọi là tham số hình thức.
  - + Các biến được khai báo để dùng riêng trong chương trình con gọi là biến cục bộ.



- + Các biến của chương trình chính được gọi là biến toàn cục.
- + Một chương trình con có thể có hoặc không có tham số hình thức, có thể có hoặc không có biến cục bộ.
- Thực hiện chương trình con: để thực hiện (gọi) một chương trình con, ta cần phải có lệnh gọi nó tương tự lệnh gọi hàm thay thủ tục chuẩn.

### III/ Cách viết và sử dụng thủ tục:

#### 1. Dạng 1: không có thông số:

```

Procedure <tên thủ tục>;
    [<phần khai báo>]
Begin
    [<dãy các lệnh>]
End;

```

- Lệnh gọi thực thi: <tên thủ tục>;

#### 2. Dạng 2: có thông số:

```

Procedure <tên thủ tục>(<danh sách tham số>);
    [<phần khai báo>]
Begin
    [<dãy các lệnh>]
End;

```

- Lệnh gọi thực thi: <tên thủ tục>(<danh sách tham số lúc thực thi>);
- Trong lệnh gọi thủ tục, các tham số hình thức được thay bằng các tham số thực tương ứng là các giá trị cụ thể được gọi là tham số giá trị (tham trị).

### IV/ Cách viết và sử dụng hàm:

```

Function <tên hàm>[(<danh sách tham số>)]:<kiểu dữ liệu trả về>;
    [<phần khai báo>]
Begin
    [<các dãy lệnh>]
    <tên hàm>:=<biểu thức>;
End;

```

- Lệnh gọi thực thi: <tên biến>:=<tên hàm>[(<danh sách tham số lúc thực thi>)];
- Chú ý: 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.

### V/ Tham số và cách truyền tham số:

- Tham số tại nơi gọi: tham số thực.
- Tham số tại nơi được gọi: tham số hình thức.

Kiểu tham số hình thức	Khai báo	Giá trị của tham số thực tương ứng sau khi thực hiện chương trình con
Tham số hình thức trị - Tham trị (truyền bằng trị)	Không có var	Giữ nguyên giá trị ban đầu trước khi gọi chương trình con.
Tham số hình thức biến – Tham biến (truyền bằng tham khảo)	Có var	Lấy giá trị kết quả của chương trình con.

- Quy tắc:

- + Quy tắc 1: số lượng tham số hình thức và số lượng tham số thực phải bằng nhau.
- + Quy tắc 2: các tham số hình thức và tham số thực phải tương ứng với nhau theo từng vị trí.
- + Quy tắc 3: các tham số hình thức và tham số thực phải tương ứng với nhau về kiểu dữ liệu.



- + Quy tắc 4: với tham số hình thức được khai báo như là tham trị thì tham số thực tương ứng có thể là hằng, biến, biểu thức, giá trị trả về từ hàm; với tham số hình thức được khai báo như là tham biến thì tham số thực tương ứng chỉ có thể là biến.
- Các quy tắc cho việc gọi chương trình con trong Pascal: Có 3 trường hợp mà một chương trình con A gọi được chương trình con B:
  - + A là cha của B.
  - + A ngang cấp với B (A và B có chung một cha).
  - + A bị bao bởi C, còn B và C ngang cấp với nhau (khi B và C trùng nhau thì B là cha của A).

**VI/ Chương trình con đề quy:**

### 1. Khái niệm đề quy:

- Một đối tượng được gọi là đệ quy nếu nó bao gồm một phần tử của chính nó hay được định nghĩa bởi chính nó.
- Một chương trình con được gọi là đệ quy nếu trong chương trình con đó có lời gọi đến chính nó.
- Có hai trường hợp:
  - + Lời gọi đệ quy trực tiếp: nếu trong một chương trình con có lời gọi tường minh đến nó: SubA  $\Rightarrow$  SubA.
  - + Lời gọi đệ quy gián tiếp: nếu trong một chương trình con A có lời gọi một chương trình con B khác, và trong B lại có lời gọi A: SubA  $\Rightarrow$  SubB  $\Rightarrow$  SubA.
- Thuật giải đệ quy thích hợp khi các bài toán, các hệ thức, các hàm, các cấu trúc dữ liệu cũng được định nghĩa theo kiểu đệ quy.

## 2. Các thành phần của chương trình đề quy:

- Sử dụng chương trình con (procedure, function) để hiện thực đệ quy.
- Hai thành phần cơ bản của đệ quy gồm:
  - + Thành phần dừng (phần neo – anchor): không chứa khái niệm đang định nghĩa;
  - + Thành phần đệ quy: có chứa khái niệm đang định nghĩa.
- Lời gọi đệ quy trong câu lệnh If:
 

**If <điều kiện dừng> then <lệnh 1> else <lệnh 2 và gọi lại chương trình con này>;**

## BÀI 2 THƯ VIÊN CHƯƠNG TRÌNH CON CHUẨN

Các Unit là những thư viện nhỏ, trong đó có chứa các chương trình con đã được tạo sẵn và người lập trình chỉ việc lấy ra sử dụng.

- **CRT:** chứa các chương trình con xử lí màn hình.
- **PRINTER:** chứa các chương trình con về in ấn.
- **DOS:** chứa các chương trình con khai thác các hàm của Dos.
- **SYSTEM:** chứa các chương trình con về dữ liệu hệ thống.
- **GRAPH:** chứa các chương trình con về đồ họa.

Goi một Unit: Uses <tên Unit>;

DOE THEM:

**Tạo Unit:**

UNIT <tên Unit>; {tên Unit thường đặt trùng với tên file}

USES <tên các Unit khác cần dùng>;

## INTERFACE

[ <phần khai báo dùng chung> ]

## IMPLEMENTATION {phần thực thi}

[<nội dung của từng chương trình con>]



END.

*BÀI TẬP*

Bài 1: Viết chương trình tính  $n!$

Giải:

Cách 1:

```

program bt;
uses crt;
var n:integer;

procedure GT(m:integer);
var i:integer; gth:int64;
begin
    if (m=0) or (m=1) then gth:=1 else
        begin
            gth:=1;
            for i:=2 to m do gth:=gth*i;
        end;
    writeln(m,'! = ',gth);
end;

begin
    clrscr;
    write('Nhập n = '); readln(n);
    GT(n);
    readln;
end.

```

Cách 2:

```

program bt;
uses crt;
var n:integer;

function GT(m:integer):int64;
var i:integer; gth:int64;
begin
    if (m=0) or (m=1) then gth:=1 else
        begin
            gth:=1;
            for i:=2 to m do gth:=gth*i;
        end;
    GT:=gth;
end;

begin
    clrscr;

```



```

write('Nhap n = '); readln(n);
writeln(n, '! = ', GT(n));
readln;

```

end.

Bài 2: Viết hàm đệ qui tính giai thừa?

Giải:

```

function GT(n:integer):int64;
begin
    if n=0 then GT:=1 else GT:=n*GT(n-1);
end;

```

Bài 3: Viết hàm đệ qui tính số Fibonacci?

Giải:

```

function F(n:integer):longint;
begin
    if n<=2 then F:=1 else F:=F(n-1)+F(n-2);
end;

```

Bài 4: Collatz đưa ra giả thiết rằng: với một số nguyên dương X, nếu X chẵn thì ta gán  $X:=X \div 2$ , nếu X lẻ thì ta gán  $X:=X*3+1$  thì sau một số hữu hạn bước, ta sẽ có  $X = 1$ . Viết chương trình nhập vào một số nguyên dương X và thực hiện giả thiết của Collatz.

Giải:

```

uses crt;
var x:integer;

procedure Slove(x:integer);
begin
    if x=1 then write(x) else
        if x mod 2=0 then
            begin
                Slove(x div 2);
                write(' *2 ');
            end else
                begin
                    Slove(x*3+1);
                    write(' div 3 ');
                end;
end;

end;

begin
    clrscr;
    write('Nhap X = '); readln(x);
    Slove(x);
    readln;
end.

```

Bài 5: Viết chương trình giải phương trình bậc nhất?

Giải:



```

program bt;
uses crt;
var a,b:integer;

procedure PTB1(a,b:integer);
begin
    if a < > 0 then writeln('x = ', -b/a:5:2) else
        if b < > 0 then writeln('PTVN') else writeln('PTVD');
end;

begin
    clrscr;
    write('Nhap he so a, b = '); readln(a,b);
    PTB1(a,b);
    readln;
end.

```

Bài 6: Viết chương trình giải phương trình bậc hai?

Giải:

```

program bt;
uses crt;
var a,b,c:integer;

procedure PTB2(a,b,c:integer);
var d:integer; x1,x2:real;
begin
    d:=sqr(b)-4*a*c;
    if d > 0 then
        begin
            x1:=(-b+sqrt(d))/(2*a);
            x2:=(-b-sqrt(d))/(2*a);
            writeln('x1 = ',x1:5:2,' x2 = ',x2:5:2);
        end else
        if d=0 then writeln('x = ', -b/(2*a):5:2) else
            writeln('PTVN');
end;

begin
    clrscr;
    write('Nhap he so a, b, c = '); readln(a,b,c);
    PTB2(a,b,c);
    readln;
end.

```

Bài 7: Viết chương trình tính  $a^n$  (a:real; n:integer).

Bài 8: Viết chương trình tạo ra máy tính con có bốn phép tính: cộng, trừ, nhân, chia (sử dụng CTC).



📖 Bài 9: Viết chương trình tính  $C_n^k = \frac{n!}{k!(n-k)!}$ .

📖 Bài 10: Viết chương trình tính  $P_n^k = \frac{n!}{(n-k)!}$ .

📖 Bài 11: Viết chương trình vẽ hình chữ nhật với các tham số chiều dài, chiều rộng và dùng chương trình để vẽ các hình chữ nhật có kích thước khác nhau.

Giải:

```
uses crt;
var a,b,i:integer;

procedure vehcn(chdai,chrong:integer);
var i,j:integer;
begin
    for i:=1 to chdai do write('*');
    writeln;
    for j:=1 to chrong-2 do
        begin
            write('*');
            for i:=1 to chdai-2 do write(' ');
            writeln('*');
        end;
    for i:=1 to chdai do write('*');
    writeln;
end;

begin
    clrscr;
    vehcn(25,10);
    writeln; writeln;
    vehcn(5,10);
    readln;
    clrscr;
    a:=4; b:=2;
    for i:=1 to 4 do
        begin
            vehcn(a,b);
            readln; clrscr;
            a:=a*2; b:=b*2;
        end;
    readln;
end.
```

📖 Bài 12: Viết chương trình rút gọn một phân số?

Giải:

```
program RGPS;
uses crt;
var tu, mau, a:integer;
```



```

function UCLN(x,y:integer):integer;
var sodu:integer;
begin
    while y<>0 do
        begin
            sodu:=x mod y;
            x:=y;
            y:=sodu;
        end;
    UCLN:=x;
end;

begin
    clrscr;
    write('Nhap tu so, mau so: '); readln(tu,mau);
    a:=UCLN(tu,mau);
    if a>1 then
        begin
            tu:=tu div a;
            mau:=mau div a;
        end;
    writeln(tu:5,mau:5);
    readln;
end.

```

Bài 13: Xây dựng một hàm có tên là LOWER(ch) để trả về kết quả là chữ cái in thường nếu ch là chữ hoa và giữ nguyên ch nếu đó là chữ thường. Hàm này là hàm ngược của hàm UPCASE.

Giải:

```

function Lower(kt:char):char;
begin
    if (kt>='A') and (kt<='Z') then Lower:=chr(ord(kt)+32);
end;

```

Bài 14: Nhập vào 2 phân số. Tính cộng, trừ, nhân, chia của hai phân số đó?

Giải:

```

program bt;
uses crt;
var a,b,c,d:integer;

procedure nhap(var tu,mau:integer);
begin
    write('Nhap vao tu so: '); readln(tu);
    write('Nhap vao mau so: '); readln(mau);
end;

procedure pheptinh(var a,b,c,d:integer);

```





```

var tu, mau: integer;
begin
    tu := a*d + b*c;
    mau := b*d;
    writeln('Tong la: ', tu, '/', mau);
    tu := a*d - b*c;
    writeln('Hieu la: ', tu, '/', mau);
    tu := a*c;
    writeln('Tich la: ', tu, '/', mau);
    tu := a*d; mau := b*c;
    writeln('Thuong la: ', tu, '/', mau);
end;

begin
    clrscr;
    nhap(a,b); nhap(c,d);
    pheptinh(a,b,c,d);
    readln;
end.

```

Bài 15: Viết chương trình giải phương trình  $x + y + z = 12$  trong phạm vi số nguyên không âm và  $x < 4$ .

Giải:

```

program bt;
uses crt;
var x,y,z: integer;

procedure giaipt;
begin
    writeln('Phuong trinh co nghiem: ');
    writeln('x':5, 'y':5, 'z':5);
    for x:=0 to 3 do
        for y:=0 to 12 do
            for z:=0 to 12 do
                if (x+y+z=12) then writeln(x:5,y:5,z:5);
end;

begin
    clrscr;
    giaipt;
    readln;
end.

```

Bài 16: Viết chương trình nhập vào một số nguyên a, kiểm tra xem a có phải là số nguyên tố hay không?

Giải:

```

program BT;

```



```
uses crt;
var a:integer;

function NT(n:integer):boolean;
var k:integer;
begin
    if n < 2 then NT:=false else
        for k:=2 to trunc(sqrt(n)) do
            if (n mod k=0) then NT:=false else NT:=true;
end;

begin
    clrscr;
    write('Nhap a = '); readln(a);
    if NT(a) then writeln(a, ' la so nguyen to.') else
        writeln(a, ' khong la so nguyen to. ');
    readln;
end.
```

Bài 17: Viết chương trình nhập vào một mảng a gồm n số nguyên, xuất ra màn hình các số chẵn, lẻ và tính tổng của chúng.

Giải:

```
program BT;
uses crt;
var n,s,i:integer; a:array[1..100] of integer;

function Chan(k:integer):boolean;
begin
    if (k mod 2=0) then Chan:=true else Chan:=false;
end;

begin
    clrscr;
    write('Nhap n = '); readln(n);
    for i:=1 to n do
        begin
            write('a[' ,i, ' ] = ');
            readln(a[i]);
        end;
    writeln('Cac so chan la: '); s:=0;
    for i:=1 to n do
        if Chan(a[i]) then
            begin
                write(a[i], ' ');
                s:=s+a[i];
            end;
    end;
```



```

        end;
    writeln;
    writeln('Tong cac so chan la: ',s);
    writeln('Cac so le la: '); s:=0;
    for i:=1 to n do
        if not Chan(a[i]) then
            begin
                write(a[i], ' ');
                s:=s+a[i];
            end;
    writeln;
    writeln('Tong cac so le la: ',s);
    readln;
end.

```

Bài 18: Viết chương trình nhập vào hai số nguyên a và b. Tính ước chung lớn nhất của chúng?  
Giải:

```

program BT;
uses crt;
var a,b:integer;

function UCLN(m,n:integer):integer;
var tmp:integer;
begin
    while n>0 do
        begin
            m:=m mod n;
            tmp:=m; m:=n; n:=tmp;
        end;
    UCLN:=m;
end;

begin
    clrscr;
    write('a, b = '); readln(a,b);
    writeln('UCLN = ',UCLN(a,b));
    readln;
end.

```

Bài 19: Viết chương trình nhập vào ba điểm Toán, Văn, Anh của một học sinh. Tính và đưa ra màn hình điểm trung bình và xếp loại.

Giai:

```

program BT;
uses crt;
var t,v,av,dtb:real;

function XL(a:real):string;

```



```
begin
    if a >= 8 then XL:='Gioi' else
        if a >= 6.5 then XL:='Kha' else
            if a >= 5 then XL:='Trung binh' else XL:='Yeu';
end;
```

```
begin
    clrscr;
    write('Nhap diem Toan, Van, Anh van: ');
    readln(t,v,av);
    dtb:=(t+v+av)/3;
    writeln('Diem trung binh: ',dtb:0:1);
    writeln('Xep loai: ',XL(dtb));
    readln;
end.
```

Bài 20: Cho tệp DATA.TXT gồm nhiều dòng, mỗi dòng chứa bộ ba số m,n,p. Viết chương trình đọc từ tệp ba số m,n,p và cho biết bộ ba số này có tạo thành một tam giác hay không (với m,n,p là độ dài ba cạnh của tam giác), nếu có thì cho biết tính chất của tam giác, tính chu vi, diện tích của tam giác đó.

Giải:

```
program BT;
uses crt;
const fi='DATA.TXT';
var m,n,p:real; f:text;

function Tamgiac(a,b,c:real):boolean;
begin
    if (a+b>c) and (a+c>b) and (b+c>a) then Tamgiac:=true else Tamgiac:=false;
end;

function Can(a,b,c:real):boolean;
begin
    if (a=b) or (a=c) or (b=c) then Can:=true else Can:=false;
end;

function Vuong(a,b,c:real):boolean;
begin
    if (a*a+b*b=c*c) or (a*a+c*c=b*b) or (b*b+c*c=a*a) then
        Vuong:=true else Vuong:=false;
end;

function Deu(a,b,c:real):boolean;
begin
    if (a=b) and (a=c) and (b=c) then Deu:=true else Deu:=false;
end;
```



```

procedure Xuli(x,y,z:real);
var ncv,s:real;
begin
    writeln('*****');
    if Tamgiac(x,y,z) then
        begin
            if Deu(x,y,z) then writeln('Tam giac deu') else
                if Vuong(x,y,z) then writeln('Tam giac vuong') else
                    if Can(x,y,z) then writeln('Tam giac can') else
                        writeln('Tam giac thuong');
            writeln('Chu vi: ',x+y+z:0:1);
            ncv:=(x+y+z)/2;
            s:=sqrt(ncv*(ncv-x)*(ncv-y)*(ncv-z));
            writeln('Dien tich: ',s:0:1);
        end else
            writeln('Khong phai la tam giac');
end;

begin
    clrscr;
    assign(f,fi); reset(f);
    while not eof(f) do
        begin
            readln(f,m,n,p);
            Xuli(m,n,p);
        end;
    close(f);
    readln;
end.

```

## MỘT SỐ BÀI TẬP TỰ LUYỆN

### I. PHẦN CĂN BẢN:

Bài 1: Viết chương trình xuất ra màn hình câu “Em sẽ là niềm vui của mẹ”.

Bài 2: Viết chương trình xuất ra màn hình hai câu:

“\*\* Đây là chương trình Pascal \*\*”

“Chúc các bạn học tập vui vẻ”

Bài 3: Viết chương trình xuất ra màn hình hình chữ nhật và hình tam giác cân sau:

```

* * * * *
*           *
*         * * *
* * * * *
* * * * *
* * * * *

```

Tiến hành đưa 2 hình chữ nhật và hình tam giác ra giữa màn hình.

Bài 4: Viết chương trình vẽ hình chữ nhật đặc ruột với chiều dài 6, chiều rộng 4 dấu “\$”. Chương trình sử dụng biến

Bài 5: Viết chương trình nhập vào hai số a, b. Tính tổng, hiệu của chúng rồi in kết quả ra màn hình.



Bài 6: Viết chương trình nhập tên, năm sinh của người sử dụng, sau đó in ra màn hình tên, tuổi của người đó.

Bài 7: Viết chương trình nhập chiều cao (cm), vòng ngực tối đa (cm), vòng ngực tối thiểu (cm), cân nặng (kg). Rồi tiến hành tính chỉ số Pignet bằng công thức:

$$\text{Pignet} = \text{chiều cao} - (\text{trung bình vòng ngực} + \text{trọng lượng})$$

Bài 8: Viết chương trình nhập vào 2 số a, b rồi tính  $a^2$ ,  $b^3$ .

Bài 9: Viết chương trình nhập vào 2 số a, b rồi thực hiện phép tính  $a \bmod b$ ,  $b \div a$ . In kết quả ra màn hình.

Bài 10: Viết chương trình tính diện tích và chu vi hình chữ nhật.

Bài 11: Viết chương trình nhập vào chu vi của hình chữ nhật rồi suy ra diện tích lớn nhất mà hình chữ nhật đó có thể có được.

Bài 12: Viết chương trình đổi inch ra cm biết  $1 \text{ inch} = 2,54 \text{ cm}$ .

Bài 13: Viết chương trình tính vận tốc khi biết quãng đường (km) và thời gian (giờ). Sau đó đổi đơn vị của vận tốc ra m/s.

Bài 14: Viết chương trình nhập vào 2 số a, b, tiến hành đổi giá trị của a cho b và ngược lại.

Bài 15: Cho nhập độ dài hai cạnh góc vuông của một tam giác. Hãy viết chương trình tính:

- Độ dài cạnh huyền.
- Chu vi diện tích của tam giác.
- Độ dài đường cao xuống cạnh huyền.

Bài 16: Viết chương trình tính diện tích và chu vi của hình tròn khi biết bán kính (sử dụng const).

Bài 17: Viết chương trình nhập tên và khối lượng của một người sau đó tính ra trọng lượng của người đó trên: Trái Đất, Mặt Trăng, Mặt Trời, Sao Thủy, Sao Mộc. Biết nếu khối lượng của trái đất là 1 thì khối lượng của mặt trời là 27,9; mặt trăng là 0,17; sao thủy là 0,37; sao Mộc là 2,64.

Bài 18: Viết chương trình nhập thời gian làm một công việc (thời gian tính bằng giây). Sau đó viết ra thời gian trên dưới dạng bao nhiêu giờ, bao nhiêu phút, bao nhiêu giây.

Bài 19: Viết chương trình nhập một STRING (chuỗi) gồm có 3 ký tự cách nhau bởi một khoảng trắng, sau đó in ra từng từ.

Bài 20: Viết chương trình nhập vào một ký tự rồi in ra mã ASCII của nó.

Bài 21: Viết chương trình nhập một số kiểu byte, sau đó in ra ký tự tương ứng của số đó trong bản mã ASCII.

Bài 22: Viết chương trình nhập vào một số nguyên dương n, sau đó tính tổng các số nguyên từ 1 đến n. (Sử dụng công thức).

## II. CẤU TRÚC ĐIỀU KIỆN VÀ CÂU LỆNH GHÉP:

Bài 23: Viết chương trình nhập tên, điểm trung bình của một học sinh. Nếu điểm trung bình lớn hơn 9 thì in ra câu "Chúc mừng bạn".

Bài 24: Viết chương trình cho nhập hai số a, b. So sánh a và b. Nếu số nào lớn hơn thì gán max bằng giá trị đó.

Bài 25: Viết chương trình nhập hai số a, b. Kiểm tra xem a và b có = 0 không? Nếu a và b cùng bằng không thì in ra câu nói "mời bạn nhập lại a và b", nếu a ≠ 0 thì tính  $a^2$ , nếu b ≠ 0 thì tính số nghịch đảo của b, còn nếu a và b cùng khác 0 thì tính  $a/b$ .

Bài 26: Nhập hai số a, b tính thương và in ra kết quả.

Bài 27: Một số nguyên chia hẳn cho 3 nếu tổng các chữ số của nó chia hẳn cho 3. viết chương trình nhập một số nguyên có tối đa 4 chữ số, xét xem số này có chia hẳn cho 3 không?

Bài 28: Viết chương trình giải phương trình bậc nhất  $ax + b = 0$  với a, b nhập từ bàn phím.

Bài 29: Viết chương trình nhập vào ba số a, b, c.

- a) In ra màn hình số nguyên lớn nhất
- b) In ra màn hình số nguyên nhỏ nhất.
- c) Sắp xếp ba số đó theo thứ tự tăng dần và giảm dần.

Bài 30: Viết chương trình nhập điểm Toán, Văn, Lý, Hoá, Anh văn (Toán, Văn hệ số 2). Tính điểm trung bình và in ra học bổng theo điều kiện ĐTB  $\geq 8$  và không có môn dưới 7 là 1.000.000 đồng. ĐTB  $\geq 6.5$  và không có môn dưới 5 là 500.000 đồng. Học bổng bằng không cho các trường hợp còn lại.

Bài 31: Viết chương trình nhập vào 3 số a, b, c bất kỳ. Kiểm tra xem ba số đó có thể là cạnh của một tam giác không. Nếu không thì in ra thông báo, nếu có thì kiểm tra xem đó là tam giác gì trong các loại tam giác sau (thường, vuông, cân, đều). Sau đó tính diện tích tam giác theo công thức

$$S = \sqrt{p(p-a)(p-b)(p-c)} \quad \text{Với } p = (a+b+c)/2$$

Bài 32: Một số tự nhiên N được gọi là Palidrom nếu cách viết thập phân của số đó là đối xứng (2112, 353). Nhập vào một số nguyên dương < 10000, hãy kiểm tra xem số đó có phải là số Palidrom không?

Bài 33: Giả sử thuế lợi tức được đánh giá trên thu nhập như sau:



Thu nhập	Thuế suất
0 → 400.000	0%
400.000 → 1.000.000	10%
1.000.000 → 2.000.000	20%
trên 2.000.000	30%

Viết chương trình hỏi thu nhập của một người, sau đó tính thuế phải nộp (sử dụng const).

Bài 34: Cho nhập quãng đường đi hãy tính số tiền Taxi phải trả theo mức giá 500 m đầu là 2500 đồng, từ km thứ 10 là 4000 đồng, từ km thứ 30 trở đi là 3000 đồng.

Bài 35: Viết chương trình nhập tọa độ (x, y) của điểm M trên mặt phẳng. Cho biết điểm M nằm ở góc phần tư nào của mặt phẳng.

### III. CÂU LỆNH CHỌN CASE .. OF

Bài 36: Viết chương trình cho nhập vào một số nguyên dương rồi in ra tên tháng tương ứng.

Bài 37: Cho biết số tháng trong năm hãy in ra số ngày tương ứng (cho tháng 2 có 29 ngày).

Bài 38: Viết chương trình nhập tháng, năm, in ra xem tháng có bao nhiêu ngày biết tháng 2 năm thường là 28 ngày, tháng 2 năm nhuận là 29 ngày. Năm nhuận là năm chia hết cho 4 hay 400 nhưng không phải năm đầu thế kỷ.

Bài 39: Viết chương trình nhập vào một số n rồi in ra giá trị của số đó bằng chữ (  $0 < n \leq 10000$  )

Bài 40: Viết chương trình cho biết một ký tự được nhập từ bàn phím thuộc nhóm ký tự nào ('A' → 'Z': ký tự hoa; 'a' → 'z': ký tự chữ thường; '0' → '9': chữ số; '+', '-', '\*', '/': toán tử).

Bài 41: Phát sinh ngẫu nhiên một số từ 0 → 9. nếu số đó nằm trong khoảng (9, 10) thì in "xuất sắc"; (7, 8) thì in "giỏi"; (5, 6) thì "khá"; (3, 4) thì "trung bình"; (1, 2) "yếu"; 0 thì kém.

Bài 42: Viết chương trình phát sinh ngẫu nhiên hai số và phát sinh ngẫu nhiên một phép toán tử rồi tính kết quả của phép toán đó.

Bài 43: Viết chương trình mô tả một máy tính với các phép toán (+, -, \*, /, căn bậc 2, bình phương, lập phương, lũy thừa một cơ số với số mũ cho trước), với 2 toán hạng và toán tử được nhập từ bàn phím.

Bài 44: Viết chương trình nhập ngày tháng năm hôm nay, in ra ngày tháng năm của ngày mai.

Bài 45: Cho giờ bắt đầu công việc và giờ kết thúc công việc. Viết chương trình tính tiền công phải trả biết rằng một giờ công buổi sáng giá 10.000 đồng (buổi sáng được tính từ 7 → 11 giờ); một giờ công buổi chiều giá 20.000 đồng (buổi chiều được tính từ 12 → 17 giờ); một giờ công buổi tối giá 40.000 đồng (buổi tối được tính từ 18 → 21 giờ).

Bài 46: Nhập chỉ số cũ, chỉ số mới và mục đích sử dụng điện. Viết chương trình tính hoá đơn tiền điện phải trả biết rằng tiền điện được tính theo giá bậc thang với các mức giá và loại giá như sau:

- Điện sinh hoạt:
  - 100 kw đầu: 550 đồng/kw.
  - 50 kw tiếp theo: 850 đồng/kw.
  - 151 → 250 kw: 1000 đồng/kw.
  - 251 kw trở lên: 1500 đồng/kw.
- Điện sản xuất:
  - 10000 kw đầu: 1500 đồng/kw
  - 10001kw trở lên: 2000 đồng/kw
- Điện nông nghiệp:
  - 700 đồng/kw
- Điện trường học và bệnh viện:
  - 800 đồng/kw
- Điện vui chơi:
  - 2500 đồng/kw.

Bài 47: Viết chương trình trò chơi "Oẳn tù tì"

- a) Hai người chơi với nhau.
- b) Một người chơi với máy.

### IV. CÂU LỆNH LẶP:

Bài 48: Viết chương trình nhập vào một số n rồi tiến hành tính tổng  $S = 1 + 2 + 3 + \dots + n$ . (ba cách)

Bài 49: Viết chương trình nhập vào một số n rồi tiến hành tính tổng  $S = 1/1 + 1/2 + \dots + 1/n$ . (ba cách)

Bài 50: Viết chương trình kiểm tra một số n được nhập từ bàn phím. Nếu  $n = 0$  thì yêu cầu người sử dụng chương trình nhập lại. (ba cách).

I. Câu lệnh lặp While – Do & Repeat – Until:



Bài 51: Viết chương trình tính tổng và tích của các số nguyên nhập từ bàn phím. Chương trình kết thúc khi nhập số 0.

Bài 52: Viết chương trình nhập từ bàn phím một số bất kỳ, kết thúc khi nhập số 0. Đếm xem các số đã nhập vào có bao nhiêu số chẵn, bao nhiêu số lẻ.

Bài 53: Viết chương trình lặp lại việc nhập một số thực bất kỳ, cho biết phần nguyên của số này có mấy chữ số. Chương trình kết thúc khi người sử dụng nhấn ESC.

Bài 54: Viết chương trình tính tổng  $S = 1/1 + 1/2 + 1/3 + \dots$  (điều kiện dừng của chương trình là  $2 - S < 0,0001$ )

Bài 55: Viết chương trình để in ra màn hình thương nguyên và dư của phép chia số nguyên dương cho 5.

## II. Câu lệnh For – To – Do:

Bài 56: Viết chương trình tính các giá trị sau (với n nhập từ bàn phím):

- $S1 = 1 + 3 + 5 + 7 + \dots + n.$
- $S2 = 1 + 2*2 + 3*3 + \dots + n*n.$
- $S3 = 1 * 2 * 3 * \dots * n$
- $S4 = 1/(1*2) + 1/(2*3) + 1/(3*4) + \dots + 1/(n*(n+1))$
- $S5 = (1 + 1/1^2)*(1 + 1/2^2)* \dots * (1 + 1/n^2)$
- $S6 = x^n/n!$

Bài 57: Viết chương trình để in ra căn bậc hai của các số chẵn từ 0  $\rightarrow$  100.

Bài 58: Viết chương trình tính N! (n giai thừa) biết ( $0! = 1$ ;  $N! = 1*2*3*\dots*(n-1)*n$  với  $n > 0$ )

Bài 59: Viết chương trình kiểm tra xem có bao nhiêu số chia hết cho 3 mà không chia hết cho 6 trong khoảng từ 120  $\rightarrow$  650.

## III. Chọn vòng lặp đúng nhất để viết chương trình:

Bài 60: Viết chương trình in ra màn hình các số có 3 chữ số với điều kiện số đó chia cho 6 dư 2 và tổng của các chữ số nó là 20.

Bài 61: Viết chương trình cho hiện lên màn hình kết quả sau:

- ```

*
* *
* * *
* * * *
* * * * *
* * * * * *

```
- ```

$ $ $ $ $ $
$ $ $ $ $
$ $ $ $
$ $ $
$ $
$

```
- ```

@
@ @ @
@ @ @ @ @
@ @ @ @ @ @
@
@ @ @ @ @ @
@ @ @
@ @ @ @ @ @
@ @ @ @

```





Bài 62: Viết chương trình in ra màn hình như sau:

PASCAL

PASCAL

PASCAL

PASCAL

Bài 63: Viết chương trình cho tạo bảng số như sau:

|     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   |
| 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 90  | 91  | 92  | 93  | 94  | 95  | 96  | 97  | 98  | 99  |

Bài 64: Nhập vào một số nguyên dương, viết chương trình in ra số đó theo chiều ngược lại. Ví dụ: 12345 → 54321.

Bài 65: Nhập số tiền phải đổi n, giả xử ngân hàng có tiền loại 500.000, 200.000, 100.000, 50.000, 20.000, 10.000, 5.000, 2.000, 1.000, 500, 200. Hãy viết chương trình đổi tiền sao cho những loại tiền có giá trị cao đổi hết rồi mới đến loại tiền có giá trị thấp. Ví dụ: 199.999 đổi được: 1 tờ 100.000, 1 tờ 50.000, 2 tờ 20.000, 1 tờ 5000, 2 tờ 2000... và thừa 99 đồng.

Bài 66: Viết chương trình cho nhập số tiền gửi, lãi suất một tháng (tính theo %), số tháng gửi. Tính tiền lãi và số tiền có được sau n tháng gửi ngân hàng.

Bài 67: Viết chương trình cho nhập số tiền gửi (a), lãi suất tháng (x). Hỏi muốn có số tiền (b) thì phải gửi bao nhiêu tháng.

Bài 68: Viết chương trình cho nhập ngày / tháng / năm. In ra xem ngày này là ngày thứ mấy. Biết ngày 1 / 1 / 1 là thứ Ba.

Bài 69: Viết chương trình cho nhập một số nguyên dương a, hãy viết lại số a bằng cách phân tích thành các thừa số nguyên tố. Ví dụ: a = 40 → in ra màn hình a = 2, 2, 2, 5.

Bài 70: Viết chương trình tìm ước chung lớn nhất của hai số nguyên dương m, n.

Bài 71: Viết chương trình tìm bội chung nhỏ nhất của hai số nguyên dương m, n.

Bài 72: Viết chương trình nhập vào một số nguyên dương rồi kiểm tra xem số đó có phải là một số hoàn hảo hay không? (số hoàn hảo là số có tổng các chữ số bằng tích các chữ số).

Bài 73: Viết chương trình nhập vào một số n (n >= 2). Kiểm tra xem số n có phải là số nguyên tố hay không?

Bài 74: Viết chương trình in ra các số nguyên tố có 2 chữ số.

Bài 75: Viết chương trình nhập vào một số n > 0, in ra tất cả các số nguyên tố nằm giữa 2 và n, đồng thời đếm xem có bao nhiêu số nguyên tố.

Bài 76: Viết chương trình in ra bản chữ chương n.

Bài 77: Viết chương trình tìm tất cả các số nguyên a, b ( 1 <= a, b <= 100 ) sao cho a<sup>2</sup> + b<sup>2</sup> là một số chính phương.

Bài 78: Viết chương trình in ra màn hình cây chữ số sau:

```

1
232
34543
4567654
567898765
67890109876
7890123210987
891023454320198
90123456765432019
0123456789876543210

```

Bài 79: Viết chương trình giả lập đàn Piano.

Hướng dẫn: Nhập ký tự từ bàn phím, quy định một ký tự là một phím đàn (Ví dụ: a → Do, s → Re...). Sử dụng hàm sound để chương trình phát ra âm thanh biết: Do → sound (532), Re → sound (578), Mi → sound (659), Fa → sound (698), Sol → sound (784), La → sound (880), Si → sound (988).

Bài 80: Viết chương trình cho xuất ra màn hình chuỗi Hello Pascal chớp tắt đổi màu. Yêu cầu: Màn hình đổi màu cùng lúc với chữ chớp tắt đổi màu.

Bài 81: Viết chương trình in ra giá trị bảng mã ASCII của 1 ký tự được nhập từ bàn phím (tất cả các phím trên bàn phím).

Bài 82: Viết chương trình nhập chiều cao của tam giác, xuất ra màn hình tam giác đặc bằng các dấu \*

Ví dụ: n = 4



Xuất

```

      *
    * * *
  * * * * *
* * * * * *

```

Bài 83: Viết chương trình nhập chiều cao của tam giác, xuất ra màn hình tam giác đặc bằng các dấu \* và dấu [

Ví dụ:  $n = 4$

Xuất

```

      *
    * [ *
  * [ * [ *
* [ * [ * [ *

```

Bài 84: Viết chương trình tính phần tử thứ  $n$  của dãy Fibonacci với  $n$  nhập từ bàn phím. Biết dãy Fibonacci được xác định như sau:

$$u_0 = u_1 = 1$$

$$u_{n+2} = u_{n+1} + u_n \quad (n > 0)$$

Ví dụ:  $n = 8 \rightarrow 1 \ 1 \ 2 \ 3 \ 5 \ 8 \ 13 \ 21$ .

Bài 85: Viết chương trình nhập một số tự nhiên lớn hơn 7. sau đó phân tích số này thành tổng tuyến tính của hai số 3 và 5. Ví dụ: Nhập 14  $\rightarrow 14 = 3*3 + 1*5$ .

Bài 86: Viết chương trình in bảng cửu chương từ 2  $\rightarrow$  9.

Bài 87: Viết chương trình in ra màn hình các số nguyên có 4 chữ số sao cho: Tổng chữ số hàng trăm + hàng ngàn = tổng chữ số hàng chục + hàng đơn vị.

Bài 88: Nhập các số nguyên dương  $a, b, c$ . Tìm các nghiệm nguyên dương của phương trình  $ax + by = c$ .

$$ax + by + cz = d$$

Bài 90 “Vừa gà vừa chó, bó lại cho tròn, ba mươi sáu con, một trăm chân chẵn”. Lập trình tính số gà, số chó.

Bài 91 Viết chương trình giải bài toán sau: “100 trâu, 100 cỏ, trâu đứng ăn 5, trâu nằm ăn 3, lụ khụ trâu già, 3 con 1 bó.

Bài 92 Có 3 loại giấy bạc 500, 200 và 50 đồng. Tìm tất cả các tổ hợp có thể có của 3 loại giấy bạc trên để cho số tiền là 5750 đồng.

Bài 93 Viết chương trình nhập và kiểm tra giá trị 2 số nguyên  $m, n$  với  $1 \leq m \leq 12$  và  $0 \leq n \leq 60$ . In ra thời gian tại thời điểm “ $m$  giờ  $n$  phút”. Hãy xác định khoảng thời gian bé nhất để hai kim đồng hồ trùng nhau kể từ thời điểm này.

Bài 94 Viết chương trình nhập 2 số  $m$  và  $n$ . tìm UCLN và BCNN. Cho biết 2 số này có nguyên tố cùng nhau không?

Bài 97 Tuổi của cha năm nay là 35, tuổi của con năm nay là 4. Hỏi sau bao nhiêu năm nữa, tuổi cha gấp đôi tuổi con.

Bài 98 Nhập các số thực  $a, b, u$  và  $c, d, v$ . Giải và biện luận hệ phương trình

$$\begin{cases} ax + by = u \\ cx + dy = v \end{cases}$$

Bài 99: Nhập hai số nguyên dương, viết chương trình kiểm tra xem hai số đó có nguyên tố cùng nhau hay không.

Bài 100: Tìm tất cả cc số thỏa điều kiện sau  $abc = a^3 \cdot b^3 \cdot c^3$

## V. CÁC BÀI TẬP SỬ DỤNG PROCEDURE VÀ FUNCTION.

Bài 100: Làm lại bài 61 dưới dạng thủ tục và hàm.

Bài 101: Cải tiến bài 100 thành các hình tam giác có thể di chuyển theo chiều ngang của màn hình cho đến khi mất hẳn.

Bài 102: Viết chương trình tính  $n!$  bằng thủ tục và hàm.

Bài 103: Cho biết kết quả của đoạn chương trình sau:

```
Program text1;
```

```
Uses crt;
```

```
Var
```

```
    a,b,c,d: integer;
```

```
Procedure P (var a,c: integer; b: integer);
```

```
Var
```

```
    d: integer;
```

```
Begin
```

```
    a:=a+1;
```

```
    b:=b+1;
```

```
    c:=c+1;
```



```
        d:=a+b+c;
End;

Begin [main]
    Clrscr;
    a:=1;
    b:=1;
    c:=1;
    d:=1;
    P(a,b,c);
    Writeln ('a= ',a,'b= ',b,'c= ',c);
    P(c,a+b,b);
    Writeln ('a= ',a,'b= ',b,'c= ',c);
    Readln;
End.

Program text2;
Uses crt;
Var
    a,b,c,d: integer;

Procedure P (var a,b: integer; c: integer);
Var
    d: integer;
Begin:
    a:=a+1;
    b:=b+1;
    c:=c+1;
    d:=a+b+c;
End;

Procedure Q (var a: integer; b: integer; var c: integer);
Begin
    a:=a+1;
    b:=b+1;
    P (b,c,a);
    c:=c+1;
    d:=d+1;
End;

Begin {main}
    Clrscr;
    a:=1;
    b:=1;
    c:=1;
    d:=1;
    P(a,b,c);
    Writeln ('a= ',a,'b= ',b,'c= ',c);
    Q(b,c,a);
    Writeln ('a= ',a,'b= ',b,'c= ',c);
    Readln;
End.
```



Bài 104: Nhập vào 2 số nguyên a, b thỏa điều kiện  $0 < m < n < 32.000$ . Viết chương trình cho biết (một câu làm trong 1 procedure hoặc Function):

- n có chia hết cho m không?
- m có là số nguyên tố không?
- Xuất ra các số nguyên tố nhỏ hơn n và lớn hơn m.

Bài 105: Viết chương trình nhập vào 2 phân số. Thực hiện các phép toán +, -, \*, / của hai phân số đó rồi in ra kết quả dưới dạng tối giản.

Bài 106: Tính tổ hợp n chập k theo công thức  $C_n^k = \frac{n!}{(n-k)! \cdot k!}$

## VI. MẢNG (ARRAY):

Bài 111: Viết chương trình theo dạng thủ tục và hàm để thực hiện các yêu cầu sau:

- Nhập mảng có n phần tử, số phần tử n được nhập từ bàn phím ( $0 \leq n \leq 100$ ).
- Xuất mảng vừa nhập ra màn hình.
- Tính trung bình cộng của các phần tử dương trong mảng. Tính trung bình nhân của các phần tử âm trong mảng.
- Tìm phần tử lớn nhất và nhỏ nhất trong mảng.
- Tìm và xuất ra màn hình các phần tử là số nguyên tố trong mảng.
- Sắp xếp mảng tăng dần.
- Sắp xếp mảng giảm dần.
- Tìm phần tử x trong mảng. Báo vị trí của phần tử đó (nếu có nhiều phần tử x thì báo nhiều vị trí).
- Nhập vào phần tử x, xóa phần tử x đầu tiên tìm thấy trong mảng. Báo vị trí của phần tử x vừa xóa.
- Xoá tất cả các phần tử x tìm thấy trong mảng, báo từng vị trí của các phần tử x đã xoá.
- Xoá từ vị trí x cho đến cuối mảng. (x nhập từ bàn phím).
- Xoá từ vị trí x cho đến đầu mảng. (x nhập từ bàn phím).
- Xoá các phần tử trong mảng khi biết vị trí và số phần tử cần xoá (xoá về cuối mảng).
- Chèn một phần tử x vào vị trí m trong mảng.
- Chèn một phần tử x vào mảng nếu phần tử đó chưa có trong mảng sau cho thứ tự của mảng không thay đổi (trước khi chèn ta cần sắp xếp mảng tăng dần hoặc giảm dần).
- Nhập một giá trị x, tìm vị trí phần tử i của mảng sao cho khoảng cách từ x đến  $A[i]$  là bé nhất.
- Nhập một giá trị nguyên p, tìm vị trí k trong dãy sao cho:  

$$A[1] + A[2] + \dots + A[k] = p * (A[k+1] + A[k+2] + \dots + A[n])$$
- Xét tính đối xứng, tính đơn điệu và tính tuần hoàn của mảng. Nếu có thì thông báo, nếu không thì sắp xếp mảng tăng dần.
- Từ mảng A ban đầu tách thành hai mảng con B và C. Mảng B chứa các số hoàn hảo, mảng C chứa các số không hoàn hảo.
- Nhập hai mảng A và B. gộp hai mảng A và B thành mảng C sao cho các phần tử của mảng B nằm trước rồi mới đến các phần tử của mảng A.
- Nhập hai mảng A và B. gộp hai mảng A và B thành mảng C sao cho các phần tử của mảng A và B nằm xen kẽ nhau.
- Sắp xếp mảng A theo dạng lẽ tăng chẵn giảm.
- Tìm các mảng con chứa các phần tử không trùng nhau từ mảng gốc ban đầu.
- Xác định số mảng con tăng và giảm trong mảng A. in ra mảng con tăng dài nhất và mảng con giảm ít nhất.
- Kiểm tra mảng B có phải là mảng con của mảng A không. Nếu có thì cho biết số lần mảng B xuất hiện trong mảng A.

## VII. MA TRẬN:

Bài 112: Viết chương trình theo dạng thủ tục và hàm để thực hiện các yêu cầu sau:

- Nhập số dòng, số cột và các phần tử của ma trận.
- Xuất ma trận.
- Tìm các phần tử lớn nhất và nhỏ nhất của ma trận. Cho biết vị trí.
- Tìm một số nhập từ bàn phím xem số đó xuất hiện trong ma trận bao nhiêu lần, cho biết vị trí. Nếu số đó không xuất hiện thì thông báo.
- Tìm và thay thế các phần tử max, min của ma trận bằng phần tử trung bình.



- f. Sắp xếp ma trận tăng dần theo hàng.
- g. Sắp xếp ma trận giảm dần theo cột.
- h. Sắp xếp ma trận giảm dần theo đường chéo chính. (Kiểm tra ma trận vuông)
- i. Sắp xếp ma trận tăng dần theo đường chéo phụ. (Kiểm tra ma trận vuông)
- j. Sắp xếp ma trận tăng dần theo hướng xoắn từ ngoài vào trong.
- k. Kiểm tra ma trận có phải là ma trận đối xứng loại một không?
- l. Kiểm tra ma trận vuông, sắp xếp các phần tử trên đường chéo chính tăng, đường chéo phụ giảm.
- m. Kiểm tra xem ma trận có bao nhiêu số có 3 chữ số, 2 chữ số và một chữ số.
- n. Kiểm tra ma trận vuông, sắp xếp các phần tử lớn nhất của mỗi hàng nằm trên đường chéo chính.
- o. Kiểm tra ma trận vuông, tính tổng giá trị tam giác vuông trên cả đường chéo.
- p. Kiểm tra ma trận vuông. Lưu đường chéo chính và đường chéo phụ vào hai mảng một chiều. Sau đó in hai mảng một chiều đó ra màn hình.
- q. Kiểm tra ma trận vuông. Lưu các đường chéo song song với đường chéo chính vào mảng một chiều theo thứ tự từ trái qua. In mảng một chiều đó.
- r. Kiểm tra ma trận có phải là ma trận đường chéo không? (các phần tử trên 2 đường chéo = 0)
- s. Kiểm tra ma trận có phải là ma trận đơn vị không? (các phần tử trên đường chéo = 1, khác = 0).
- t. Kiểm tra ma trận có phải là ma trận thưa không? (hơn nửa số phần tử bằng không).
- u. Kiểm tra ma trận có phải là ma trận tam giác không? (các phần tử trên đường chéo chính = 0).
- v. Kiểm tra tính đối xứng tâm, đối xứng chéo chính, đối xứng chéo phụ của ma trận.
- w. Kiểm tra ma trận có phải là ma trận "La Tinh" hay không? (mỗi hàng, mỗi cột đều là hoán vị của [1..n])
- x. Kiểm tra ma trận có phải là ma phương hay không? (tổng mỗi hàng = tổng mỗi cột = tổng mỗi đường chéo).
- y. Xoá các phần tử trùng nhau trong ma trận. (nếu vì xoá mà ma trận không đủ hàng, cột thì ta thêm vào các phần tử lớn nhất của ma trận sao cho đủ hàng, cột).
- z. Xoá hàng, cột của ma trận.
- aa. Nhập 3 mảng A, B, C rồi tiến hành tạo ma trận từ 3 mảng trên (số dòng 3, số cột = tổng các phần tử trong 3 mảng chia ba. Nếu thiếu thì bổ sung thêm các phần tử bé nhất trong ba mảng cho đủ).
- bb. Chèn hàng, cột mới vào ma trận.
- cc. Tính tổng hai ma trận.
- dd. Tính tích hai ma trận.
- ee. Lập ma trận chuyển vị.
- ff. Nhập vào số a ( $a < n$  và  $a < m$ ). In ma trận tích a.a ra màn hình. Tính từ vị trí [1,1].
- gg. Lập ma phương với n lẻ gồm các số nguyên từ 1 đến n.n.

## VIII. CHUỖI KÝ TỰ (STRING):

Bài 113: Viết chương trình theo dạng thủ tục và hàm để thực hiện các yêu cầu sau:

- a. Nhập chuỗi ký tự từ bàn phím.
- b. Xuất chuỗi ký tự ra màn hình.
- c. Tự viết lại các hàm, các thủ tục xử lý chuỗi của pascal.
- d. Viết hàm LTRIM(s): cắt bỏ các khoảng trống bên trái chuỗi.
- e. Viết hàm RTRIM(s): cắt bỏ các khoảng trống bên phải chuỗi.
- f. Viết hàm TRIM(s): cắt bỏ các khoảng trống bên phải và bên trái chuỗi.
- g. Viết hàm WTRIM(s): cắt bỏ khoảng trống dư giữa các từ.
- h. Viết hàm S\_UPPER(s): biến toàn bộ chuỗi thành chữ hoa.
- i. Viết hàm S\_LOWER(s): biến toàn bộ chuỗi thành chữ thường.
- j. Viết hàm S\_PROPER(s): biến toàn bộ từ đầu của chuỗi thành chữ hoa.
- k. Viết hàm S\_TRUNC(s,n,m): trả về chuỗi mới sau khi cắt bỏ m ký tự kể từ vị trí n trong chuỗi gốc s.
- l. In chuỗi đảo ngược.
- m. Kiểm tra chuỗi có là chuỗi đối xứng không? Nếu không đối xứng thì đảo chuỗi.
- n. Tính tổng các chữ số trong chuỗi.
- o. Đếm các ký tự hoa, thường, số và các ký tự khác trong chuỗi.
- p. Kiểm tra trong chuỗi có k ký tự kề nhau mà như nhau hay không?

Bài 114: Nhập hai xâu ký tự. Đếm số lần xuất hiện của xâu này trong xâu kia.

Bài 115: Nhập n ký tự sao không có hai ký tự trùng nhau (nếu trùng nhau yêu cầu nhập lại).

Bài 116: Nhập một số nguyên và một số k. xoá đi k chữ số trong số nguyên đó sao cho số còn lại có giá trị bé nhất.

Bài 117: Lập trình đếm số lần xuất hiện của mỗi loại ký tự trong một chuỗi ký tự. Xuất ra màn hình các ký tự và số lần xuất hiện của nó.



## IX. LẬP VẬT:

Bài 121: Bảng mã ASCII:

+ Để biết được chữ cái mình muốn là số nào thì dùng lệnh ord(tên biến nhập vào) để chuyển từ 1 chữ sang một số tương ứng trong bảng mã. (đặt biến theo kiểu char)

+ Để biết được số mình muốn là chữ cái nào thì dùng lệnh chr(tên biến nhập vào) để chuyển từ 1 số sang một chữ tương ứng trong bảng mã. (đặt biến theo kiểu integer; longint; real hoặc byte)

Bài 122: Cho nhập vào hai phân số sau đó tính ra tổng, hiệu, tích, thương của chúng, rút gọn nếu được.

Bài 123: Cho nhập vào hai phân thức sau đó tính ra tổng, hiệu, tích, thương của chúng, rút gọn nếu được.

Bài 124: Cho nhập vào một phân thức sau đó rút gọn phân thức đó nếu được, không thì phân tích nhân tử cả tử và mẫu.

Bài 125: Cho nhập vào một số, xét xem số đó có phải là số nguyên tố không.

## MỘT SỐ HÀM VÀ THỦ TỤC

### 1. Viết thủ tục vẽ hình chữ nhật khi biết chiều dài và chiều rộng:

```

procudre HCN(dai,rong:integer);
var i,j:integer;
begin
    for i:=1 to dai do write('*'); writeln;
    for j:=1 to rong-2 do
        begin
            write('*');
            for i:=1 to dai-2 do write('*'); writeln('*');
        end;
    for i:=1 to dai do write('*'); writeln;
end;
```

### 2. Viết thủ tục hoán đổi hai số nguyên:

```

procedure Swap(var x,y:integer);
var tmp:integer;
begin
    tmp:=x; x:=y; y:=tmp;
end;
```

### 3. Viết thủ tục giải phương trình bậc nhất $ax+b=0$ :

```

procedure PTB1(a,b:real);
begin
    if a=0 then
        if b=0 then writeln('PT co vo so nghiem') else writeln('PT vo nghiem')
        else writeln('PT co nghiem x = ', -b/a:4:2);
    end;
```

### 4. Viết thủ tục giải bất phương trình bậc nhất:

```

procedure BPT1(a,b:real);
begin
    if a < > 0 then
        if a > 0 then writeln('BPT co nghiem x >= ', -b/a:4:2) else
            writeln('BPT co nghiem x <= ', -b/a:4:2) else
            if b >= 0 then writeln('BPT co vo so nghiem') else
                writeln('BPT vo nghiem');
    end;
```

### 5. Viết thủ tục giải phương trình bậc hai:

```

procedure PTB2(a,b,c:real);
```



```

var d:real;
begin
    if a=0 then
        if b=0 then
            if c=0 then writeln('PT co vo so nghiem') else writeln('PT vo nghiem') else
                writeln('PT co nghiem x = ', -c/b:4:2) else
        begin
            d:=b*b-a*a*c;
            if d=0 then writeln('PT co nghiem kep x = ', -b/(2*a):4:2) else
                if d<0 then writeln('PT vo nghiem') else
                    writeln('PT co hai nghiem x1 = ', (-b+sqrt(d))/(2*a):4:2, ' ; x2 = ', (-b-
                    sqrt(d))/(2*a):4:2);
        end;
    end;
end;

```

#### 6. Viết hàm tìm ước chung lớn nhất của hai số:

```

function UCLN(a,b:integer):integer;
var tmp:integer;
begin
    if b>0 then
        begin
            a:=a mod b;
            tmp:=a; a:=b; b:=tmp;
        end;
    UCLN:=a;
end;

```

#### 7. Viết hàm tính giai thừa của một số nguyên:

```

function GT(n:integer):extended;
begin
    if n=0 then GT:=1 else GT:=n*GT(n-1);
end;

```

#### 8. Viết hàm tính số Fibonacci:

```

function Fibo(n:integer):longint;
begin
    if n<=2 then Fibo:=1 else Fibo:=Fibo(n-1)+Fibo(n-2);
end;

```

#### 9. Viết hàm xếp loại học lực dựa vào điểm trung bình:

```

function XL(a:real):string;
begin
    if a>=8 then XL:='Gioi' else
        if a>=6.5 then XL:='Kha' else
            if a>=5 then XL:='Trung Binh' else XL:='Yeu';
end;

```

#### 10. Viết hàm trả về chữ cái in thường nếu đó là chữ in hoa, và giữ nguyên nếu đó là chữ in thường (hàm ngược của hàm UPCASE)

```

function LOWER(kt:char):char;
begin
    if (kt>='A') and (kt<='Z') then
        LOWER:=chr(ord(kt)+32) else LOWER:=kt;
end;

```



**11. Viết hàm kiểm tra số nguyên tố:**

```
function NT(n:integer):boolean;
var k:integer;
begin
    if n = 1 then exit(false);
    for k := 2 to trunc(sqrt(n)) do
        if (n mod k = 0) then exit(false);
    exit(true);
end;
```

**12. Viết hàm tính tổng các chữ số của một số nguyên:**

```
function Tongcs(n:integer):integer;
var s:integer;
begin
    s := 0;
    while n <> 0 do
        begin
            s := s + n mod 10; n := n div 10;
        end;
    tongcs := s;
end;
```

**13. Viết hàm tính tổng các ước số của một số nguyên:**

```
function tongus(n : integer): integer;
var i, s : integer;
begin
    s := 0;
    for i := 1 to n div 2 do
        if n mod i = 0 then s := s + i;
    tongus := s;
end;
```

**14. Viết hàm tính hàm mũ:**

```
function hammu(a : real; n : integer): real;
var s : real; i : integer;
begin
    s := 1;
    for i := 1 to n do s := s * a;
    hammu := s;
end;
```

**15. Viết hàm đếm số từ của một chuỗi:**

```
function sotu(s:string):integer;
var i, dem : integer;
begin
    s := ' ' + s; dem := 0;
    for i := 2 to length(s) do
        if (s[i-1] = ' ') and (s[i] <> ' ') then dem := dem + 1;
    sotu := dem;
end;
```

**16. Viết hàm kiểm tra chuỗi đối xứng:**

```
function doixung(x:string):boolean;
var y : string; i : integer;
```





```
begin
  y := "";
  for i := length(x) downto 1 do y := y + x[i];
  if x=y then doixung := true else doixung := false;
end;
```

### 17. Viết hàm tính $\sqrt[n]{x}$ :

```
function Can(n,x:integer):real;
begin
  can:=exp((1/n)*ln(x));
end;
```

## PHẦN BỔ SUNG

### I/ Một số phím chức năng thường dùng:

- **F2:** Lưu chương trình đang soạn thảo vào đĩa.
- **F3:** Mở file mới hoặc file đã tồn tại trên đĩa để soạn thảo.
- **Alt-F3:** Đóng file đang soạn thảo.
- **Alt-F5:** Xem kết quả chạy chương trình.
- **F8:** Chạy từng câu lệnh một trong chương trình.
- **Alt-X:** Thoát khỏi Free Pascal.
- **Alt-<Số thứ tự của file đang mở>:** Dịch chuyển qua lại giữa các file đang mở.
- **F10:** Vào hệ thống Menu của Pascal.

### II/ Các thao tác cơ bản khi soạn thảo chương trình:

#### 1. Các phím thông dụng:

- **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.
- **Del:** Xóa ký tự ngay tại vị trí con trỏ.
- **Back Space (←):** Xóa ký tự bên trái con trỏ.
- **Ctrl-PgUp:** Đưa con trỏ về đầu văn bản.
- **Ctrl-PgDn:** Đưa con trỏ về cuối văn bản.
- **Ctrl-Y:** Xóa dòng tại vị trí con trỏ.

#### 2. Các thao tác trên khối văn bản:

- Chọn khối văn bản: **Shift + <Các phím ←↑→↓>**
- **Ctrl-KY:** Xóa khối văn bản đang chọn
- **Ctrl-Insert:** Đưa khối văn bản đang chọn vào Clipboard
- **Shift-Insert:** Dán khối văn từ Clipboard xuống vị trí con trỏ.

### III/ Các hàm số học sử dụng cho kiểu số nguyên và số thực:

- **SQR(x):** Trả về  $x^2$
- **SQRT(x):** Trả về căn bậc hai của x ( $x \geq 0$ )
- **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  $\arctan(x)$  theo radian



- **LN(x):** Trả về  $\ln(x)$
- **EXP(x):** Trả về  $e^x$
- **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 ( $n:=n-1$ ).

#### IV/ 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**: Xóa 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**: Xóa 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  $color \in [0,15]$ .
- Thủ tục **TEXTBACKGROUND(color:Byte)**: Thiết lập màu nền cho màn hình.

### *ĐỌC THÊM.*

### **ĐỆ QUI**

#### **1. Khái niệm đệ qui:**

Trong một chương trình, một CTC có thể gọi một CTC khác vào làm việc. Nếu như CTC đó *gọi lại chính nó* thì gọi là sự đệ qui.

#### **2. Phương pháp thiết kế giải thuật đệ qui:**

- Tham số hóa bài toán.
- Tìm trường hợp suy biến.
- Phân tích các trường hợp chung (đưa về các bài toán cùng loại nhưng nhỏ hơn).

Ví dụ: Viết hàm đệ qui để tính  $n! = 1.2...n$ .

- Tham số hóa:  $n! = \text{Factorial}(n)$ ;
- $\text{Factorial}(0) = 1$  (trường hợp suy biến)
- $\text{Factorial}(n) = n * \text{Factorial}(n-1)$  (trường hợp chung)

Function **Factorial**(N:integer):Longint;

Begin

If N=0 Then **Factorial**:=1

Else **Factorial**:=N\***factorial**(N-1); { lời gọi đệ qui }

End;

#### **3. Giải thuật quay lui:**

##### **Bài toán:**

Hãy xây dựng các bộ giá trị gồm n thành phần  $(x_1, ..., x_n)$  từ một tập hữu hạn cho trước sao cho các bộ đó thỏa mãn yêu cầu B cho trước nào đó.

##### **Phương pháp chung**

Giả sử đã xác định được k-1 phần tử đầu tiên của dãy:  $x_1, ..., x_{k-1}$ . Ta cần xác định phần tử thứ k. Phần tử này được xác định theo cách sau:

- Giả sử  $T_k$ : tập tất cả các giá trị mà phần tử  $x_k$  có thể nhận được. Vì tập  $T_k$  hữu hạn nên ta có thể đặt  $n_k$  là số phần tử của  $T_k$  theo một thứ tự nào đó, tức là ta có thể thành lập một ánh xạ 1-1 từ tập  $T_k$  lên tập  $\{1, 2, ..., n_k\}$ .



- Xét  $j \in \{1, 2, \dots, n_k\}$ . Ta nói rằng “***j* chấp nhận được**” nếu ta có thể bổ sung phần tử thứ  $j$  trong  $T_k$  với tư cách là phần tử  $x_k$  vào trong dãy  $x_1, \dots, x_{k-1}$  để được dãy  $x_1, \dots, x_k$ .

- Nếu  $k=n$ : Bộ  $(x_1, \dots, x_k)$  thỏa mãn yêu cầu B, do đó bộ này được thu nhận.

- Nếu  $k < n$ : Ta thực hiện tiếp quá trình trên, tức là phải bổ sung tiếp các phần tử  $x_{k+1}$  vào dãy  $x_1, \dots, x_k$ .

Sau đây là thủ tục đệ qui cho giải thuật quay lui:

Procedure TRY( $k$ :Integer);

Var  $j$ :Integer;

Begin

For  $j := 1$  To  $n_k$  Do

If  $\langle j \text{ chấp nhận được} \rangle$  Then

Begin

$\langle$  Xác định  $x_k$  theo  $j \rangle$ ;

If  $k=n$  Then  $\langle$  Ghi nhận một bộ giá trị  $\rangle$

Else TRY( $k+1$ ); { Quay lui }

End;

End;

## ĐỌC THÊM

## FILE KHÔNG ĐỊNH KIỂU (FILE VẬT LÝ)

### 1. Khái niệm:

File không định kiểu là file không xác định kiểu của mỗi thành phần trong file, mà được hiểu là một dãy byte, mỗi phần tử có kích thước  $k$  byte, quy định bởi người lập trình. File không định kiểu tương hợp với mọi kiểu file.

### 2. Khai báo:

Var  $\langle$  Tên biến File  $\rangle$ : File;

### 3. Các thủ tục và hàm có thể thao tác trên file không định kiểu:

#### 3.1. Mở file:

Mở file chưa có trên đĩa: **Rewrite(F, k);**

Mở file đã có trên đĩa: **Reset(F, k);**

Giá trị  $k$  mô tả số lượng byte sẽ được đọc ghi trong một thao tác. Kích thước của file phải là bội số của  $k$ .

#### 3.2. Xuất/ nhập dữ liệu:

Cú pháp: **BlockRead(F, x, n [,Kq]);**

**BlockWrite(F, x, n [,Kq]);**

Chức năng:

- Đọc/ Ghi  $n$  “bản ghi”. Mỗi “bản ghi” được hiểu là một phần tử  $k$  byte.
- $x$  chứa nội dung đọc/ghi
- $Kq$  là số lượng “bản ghi” được thực hiện.

**Chú ý:** File không định kiểu thường được dùng trong các thao tác sao chép với tốc độ cao.

## ĐỌC THÊM

## DỮ LIỆU KIỂU CON TRỎ

### I. KHAI BÁO

Type

$\langle$  Tên kiểu con trỏ  $\rangle = \wedge \langle$  Kiểu của biến động  $\rangle$ ;

Var

$\langle$  Tên biến  $\rangle : \langle$  Tên kiểu con trỏ  $\rangle$ ;

#### Ví dụ 1:

Type

TroNguyen :  $\wedge$  integer;

Var

$p, q$ : TroNguyen;

Sau khai báo này các biến  $p$  và  $q$  là các biến con trỏ có thể trỏ đến các biến động có kiểu integer. Chương trình sẽ cấp phát 4 byte cho mỗi biến con trỏ. Còn vùng nhớ của các biến động chưa được cấp phát.



Ví dụ 2:

```

Type
    TroSv = ^ Sinhvien;
    Sinhvien = Record
        Hoten: String[20];
        Diem: real;
        Tiep: TroSv;
    End;

Var
    p: TroSv;
  
```

Trong ví dụ này, p là biến trỏ có thể trỏ đến các bản ghi có kiểu Sinhvien, trong bản ghi này lại có trường Tiep là một biến trỏ có thể trỏ đến biến động khác cũng có kiểu Sinhvien.

## II. LÀM VIỆC VỚI BIẾN ĐỘNG

### 2.1. Cấp phát vùng nhớ

Dùng thủ tục New theo cú pháp:

New(<biến trỏ>);

Phép gán giữa hai biến trỏ được thực hiện nếu chúng có cùng kiểu. Sau phép gán p:=q; các con trỏ p và q cùng trỏ đến một địa chỉ. Do đó mọi thay đổi của p<sup>^</sup> cũng làm thay đổi q<sup>^</sup>. Như vậy, cần phân biệt hai phép gán p:=q và p<sup>^</sup>:=q<sup>^</sup>. Ngoài ra, các con trỏ cùng kiểu có thể được so sánh với nhau bằng các toán tử quan hệ = và <>.

Turbo Pascal cũng khai báo sẵn một con trỏ không trỏ tới một biến động nào gọi là con trỏ Nil. Giá trị con trỏ Nil là tương hợp với mọi kiểu con trỏ. Nil có thể được gán cho biến con trỏ để chỉ ra rằng con trỏ ấy hiện không được sử dụng. Chúng ta cũng có thể sử dụng Nil trong các phép so sánh.

### 2.2. Giải phóng vùng nhớ

Dùng thủ tục Dispose(p);

Trong đó p là một biến con trỏ. Thủ tục Dispose cho phép trả lại bộ nhớ động đã được cấp phát bởi thủ tục New.

## III. DANH SÁCH ĐỘNG

### 3.1. Khái niệm

Chúng ta đã từng làm quen với kiểu mảng, lưu danh sách gồm nhiều thành phần có cùng kiểu. Mỗi thành phần là một biến tĩnh và số lượng thành phần của danh sách là cố định. Ở đây chúng ta đề cập đến một dạng danh sách động theo nghĩa: mỗi thành phần là một biến động và số lượng thành phần của danh sách có thể thay đổi. Mỗi biến động trong danh sách được gọi là một nút.

### 3.2. Khai báo

Để khai báo một danh sách động trước hết ta khai báo kiểu của mỗi nút trong danh sách.

```

Type <Trỏ nút> = ^ <Nút>;
    <Nút> = Record
        Data: DataType;
        Next: <Trỏ Nút>;
    End;
  
```

Var First: <Trỏ Nút>;

First là địa chỉ của nút đầu tiên trong danh sách, dựa vào trường Tiep của nút này ta biết được địa chỉ của nút thứ hai, cứ như vậy ta biết được địa chỉ của tất cả các nút trong danh sách. Danh sách dạng này được gọi là danh sách liên kết đơn.

### 3.3. Các thao tác thường gặp trên danh sách liên kết đơn

Trong phần này chúng ta giả thiết rằng mỗi nút trong danh sách có hai trường: trường Info (lưu nội dung của biến động) và trường Next (lưu địa chỉ của nút tiếp theo). ta có khai báo danh sách như sau

```

Type TroNut = ^ Nut;
    Nut = Record
        Info: data; {data là kiểu dữ liệu đã định nghĩa trước}
        Next: TroNut;
    End;
    Var First: TroNut;
  
```

#### 3.3.1. Khởi tạo danh sách



First: = Nil;

### 3.3.2. Bổ sung một nút vào đầu danh sách

{1. Tạo ra nút mới}

New(p);

p ^ .Info: = X;

{2. Bổ sung vào đầu danh sách}

p ^ .Next: = First;

First: = p;

### 3.3.3. Bổ sung một nút vào cuối danh sách

Xuất phát danh sách không có nút nào cả. Nút mới thêm vào sẽ nằm cuối danh sách. Khi đó ta cần hai biến con trỏ First và Last lượt trỏ đến các nút đầu và cuối danh sách.

Procedure Khoitao;

var p: TroNut;

Begin

First: = nil; Last: = nil;

While < còn thêm nút mới vào danh sách > do

Begin

New(p);

Readln(p ^ .Info);

p ^ .Next: = Nil;

If First = Nil then

First: = p

Else

Last ^ .next: = p;

Last: = p;

End;

End;

### 3.3.4. Duyệt danh sách

Duyệt danh sách là thăm và xử lý từng nút trong danh sách.

Procedure Duyệt;

Var p: Tronut;

Begin

p: = First;

While p <> nil do

Begin

<Xử lý p>;

p: = p ^ .Next; {duyet qua nút tiếp theo}

End;

End;

### 3.3.5. Bổ sung một nút vào sau nút được trỏ bởi p

Thủ tục sau thực hiện việc bổ sung một nút có nội dung x vào sau nút được trỏ bởi p.

Procedure Bosung(p,x);

Var q: TroNut;

Begin

New(q);

q ^ .info: = x;

if first = nil then

begin

q ^ .next : = nil;

first : = q;

end

else

begin



```

      q^.next:= p^.next;
    p^.next:= q;
    end;

```

End;

### 3.3.6. Xóa một nút khỏi danh sách

Thủ tục sau thực hiện việc xóa một nút trở bởi p ra khỏi danh sách.

```

Procedure Xoa(p);
Var    q: TroNut;
Begin
    if First = nil then
        exit;
    if p = First then
        First := First^.next
    else
        begin
            q:= First;
            While q^.next <> p do
                q:= q^.next;
            q^.next:= p^.next;
        end;
    Dispose(p);
End;

```

*ĐỌC THÊM:*

## ĐỒ HỌA

### I. MÀN HÌNH TRONG CHẾ ĐỘ ĐỒ HỌA ( GRAPHIC)

Hình ảnh trong chế độ đồ họa được tạo ra bằng các điểm ảnh (Pixel), số điểm ảnh của màn hình đồ họa tùy thuộc vào từng loại CARD màn hình và MODE qui định cho màn hình đó.

Việc lập trình trong chế độ đồ họa cần phải xác định được loại màn hình đang sử dụng và chương trình phải vận hành được trên nhiều loại màn hình khác nhau.

Tọa độ của một điểm ảnh trên màn hình đồ họa cũng giống như trong chế độ văn bản (TEXT) với điểm ảnh đầu tiên trên góc trái màn hình là (0,0), tọa độ đỉnh dưới phải tùy thuộc vào độ phân giải của màn hình, CARD màn hình và MODE màn hình.

(0,0)



(MaxX,MaxY)

Để sử dụng được chế độ đồ họa trên màn hình, ta cần phải có các File sau:

|           |                      |
|-----------|----------------------|
| GRAPH.TPU | Chứa các lệnh đồ họa |
| * .BGI    | Chứa Font màn hình   |
| * .CHR    | Chứa Font ký tự      |

## II. KHỞI TẠO VÀ THOÁT KHỎI CHẾ ĐỘ ĐỒ HỌA

### 2.1. Khởi tạo chế độ đồ họa

Thủ tục **INITGRAPH**(Gd,Gm:Integer; Path:String);

trong đó:

- Gd: Chỉ CARD màn hình.

Thông thường, một chương trình phải được chạy trên nhiều loại màn hình khác nhau nên ta có thể khai báo:

Gd = Detect                      (= 0 )

Với hằng Detect, máy sẽ tự động tìm CARD màn hình tương ứng để chạy chương trình.

- Gm: Chỉ MODE màn hình.



Trong trường hợp khai báo  $Gd = \text{Detect}$  thì không cần thiết phải khai báo  $Gm$  vì máy tính sẽ tự xác định loại CARD màn hình và thiết lập chế độ MODE màn hình tương ứng với CARD màn hình đó.

- Path: Đường dẫn đến nơi chứa các file \*.BGI. Nếu Path = '' thì ta hiểu là các file \*.BGI nằm trong thư mục hiện hành.

Hàm **GRAPHRESULT**:Integer;

Hàm này trả về kết quả của việc khởi động đồ họa.

= 0 : Thành công.

<>0 : Bị lỗi.

Tên của lỗi được xác định bởi hàm **GRAPHERRORMSG**(Er:Integer):String;

Hàm này cho ra một xâu ký tự thông báo lỗi của đồ họa xác định bởi đối số Er.

\* Hằng số GrOK = 0: Việc khởi động đồ họa có lỗi.

Ví dụ:

Uses Graph;

Procedure ThietLapDoHoa;

var gd,gm,Gr:integer;

Begin

DetectGraph(Gd,Gm);

InitGraph(gd,gm,'C:\TP\BGI');

Gr:=GraphResult;

If Gr<>GrOK then

Begin

writeln('Lỗi Do họa: ',GraphErrorMsg(Gr));

Halt(1);

End;

End;

BEGIN

ThietLapDoHoa;

...

END.

**Chú ý:** Ta có thể khởi tạo mode đồ họa với chế độ 256 màu bằng cách sử dụng hàm **InstallUserDriver**(Name:String;Ptr:Pointer):Integer; với điều kiện trên đĩa phải có file **SVGA256.BGI**.

Procedure ThietLapDoHoa;

var gd,gm,Gr:integer;

Begin

Gd:= InstallUserDriver(' SVGA256' ,NIL);

Gm:=2; {Mode 640x480x256}

InitGraph(gd,gm,'C:\TP\BGI');

End;

## 2.2. Thoát khỏi chế độ đồ họa

Thủ tục **CLOSEGRAPH**;

Sau đây là cấu trúc chung của một chương trình đồ họa:.

Uses Crt,Graph;

Procedure ThietLapDoHoa;

var gd,gm,Gr:integer;

Begin

DetectGraph(Gd,Gm);

InitGraph(gd,gm,'C:\TP\BGI');

Gr:=GraphResult;

If Gr<>GrOK then

Begin

writeln('Lỗi Do họa: ',GraphErrorMsg(Gr));

Halt(1);

End;

End;

End;



BEGIN

ThietLapDoHoa;

...

CloseGraph;

END.

### III. TỌA ĐỘ VÀ CON TRỎ TRÊN MÀN HÌNH ĐỒ HỌA

#### 3.1. Hàm **GetMaxX:Integer**;

Cho tọa độ cột lớn nhất của màn hình.

#### 3.2. Hàm **GetMaxY:Integer**;

Cho tọa độ dòng lớn nhất của màn hình.

#### 3.3. Thủ tục **MOVETO(x,y:Integer)**;

Di chuyển con trỏ từ vị trí đang đứng đến tọa độ (x,y).

#### 3.4. Thủ tục **MOVEREL(dx,dy:Integer)**;

Di chuyển con trỏ từ vị trí đang đứng đến tọa độ mới cách tọa độ cũ khoảng cách là dx, dy.

#### 3.5. Vẽ một điểm trên màn hình:

Dùng thủ tục **PUTPIXEL(x,y:Integer; color:Word)**;

#### 3.6. Lấy màu của một điểm tại tọa độ x,y:

Hàm **GETPIXEL(x,y:Integer):Word**;

### IV. ĐẶT MÀU TRÊN MÀN HÌNH ĐỒ HỌA

#### 4.1. Đặt màu cho đối tượng cần vẽ

Dùng thủ tục **SETCOLOR(Color:Byte)**;

#### 4.2. Đặt màu nền

Dùng thủ tục **SETBKCOLOR(Color:Byte)**;

### V. CỬA SỐ TRONG CHẾ ĐỘ ĐỒ HỌA

#### 5.1. Đặt cửa sổ trên màn hình

Thủ tục **SETVIEWPORT(x1,y1,x2,y2:Integer; Clip:Boolean)**;

Với x1,y1: đỉnh trên trái của cửa sổ.

x2,y2: đỉnh dưới phải của cửa sổ.

Clip = TRUE: những gì vượt khỏi màn hình sẽ bị cắt bỏ.

Clip = FALSE: những gì vượt khỏi màn hình sẽ không bị cắt bỏ.

\* Khi tạo cửa sổ thì tọa độ trên màn hình sẽ thay đổi theo.

Tọa độ mới = Tọa độ cũ - Tọa độ đỉnh trên trái.

#### 5.2. Xóa hình ảnh trong cửa sổ

- Xóa hình ảnh trong cửa sổ, ta dùng thủ tục **CLEARVIEWPORT**;

- Xóa toàn bộ màn hình, ta dùng thủ tục **CLEARDEVICE**;

### VI. VIẾT CHỮ TRONG CHẾ ĐỘ ĐỒ HỌA

#### 6.1. Chọn Font chữ

Ta dùng thủ tục **SETTEXTSTYLE(font,Dir,size:Word)**;

- Các font có thể chứa các hằng sau:

DefaultFont = 0;      TriplexFont = 1;      SmallFont = 2;

SansSerifFont = 3;      GothicFont = 4;

- Dir có các hằng sau:

HorizDir = 0      Từ trái qua phải.

VetDir = 1      Từ dưới lên trên.

- Size: độ lớn của chữ.

#### 6.2. Chọn phân bố chữ

Dùng thủ tục **SETTEXTJUSTIFY(Hz,Vt:Word)**;

Chọn vị trí của chữ xung quanh tọa độ định sẵn.

- Hz là phân bố chữ theo trục ngang. Có các hằng sau:

LeftText = 0      Chữ viết nằm bên phải trục đứng.

CenterText = 1      Chữ viết nằm ở giữa trục đứng.

RightText = 2      Chữ viết nằm bên trái trục đứng.

- Vt là bố trí chữ theo hướng dọc đối với tọa độ qui định xuất chuỗi. Các hằng liên quan:

BottomText = 0      Chữ viết nằm bên trên trục ngang.

CenterText = 1      Chữ viết nằm ở giữa trục ngang.





TopText = 2                      Chữ viết nằm bên dưới trục ngang.

### 6.3. Viết một chuỗi ký tự lên màn hình

- Xuất một chuỗi ký tự tại vị trí con trỏ:  
Dùng thủ tục OUTTEXT(St:String);
- Xuất một chuỗi ký tự tại tọa độ x,y:  
Dùng thủ tục OUTTEXTXY(x,y:Word; St:String);

Chú ý: Cách xuất chuỗi của hai thủ tục trên được qui định trong thủ tục SETTEXTJUSTIFY và SETTEXTSTYLE.

## VII. VẼ CÁC HÌNH CƠ BẢN

### 7.1. Chọn kiểu đường

- Dùng thủ tục SETLINESTYLE(Ls,Pt,Tk:Word);
- Thủ tục này xác định kiểu đường được vẽ trong đồ họa.
- Ls: kiểu đường vẽ. Ls có các giá trị sau:
  - 0: Đường liền nét
  - 1: Nét đứt
  - 2: Nét chấm gạch
  - 3: Nét gạch
  - 4: Đường do người thiết kế tạo ra.
- Pt: xác định màu vẽ.
  - . Nếu Ls = 0..3 thì Pt=0 (Lấy giá trị Default)
  - . Nếu Ls = 4 thì Pt là số nguyên chỉ màu của kiểu đường.
- Tk: xác định độ dày của đường.
  - Tk = 1: bình thường.
  - Tk = 3: đậm nét.

### 7.2. Vẽ đoạn thẳng

- LINE(x1,y1,x2,y2:Integer); vẽ từ điểm (x1,y1) đến điểm (x2,y2)
- LINETO(x,y:Integer); vẽ từ vị trí con trỏ đến điểm (x,y)
- LINEREL(dx,dy:Integer); vẽ từ vị trí con trỏ đến điểm cách đó một khoảng dx,dy.

### 7.3. Vẽ hình chữ nhật

- Dùng thủ tục RECTANGLE(x1,y1,x2,y2:Integer);

### 7.4. Vẽ cung tròn

- Thủ tục ARC(x,y:Integer; g1,g2,R:Word);
- Vẽ cung tròn có tâm (x,y) bán kính R, góc bắt đầu là g1 và góc kết thúc là g2.

### 7.5. Vẽ đường tròn - Ellip

- Thủ tục vẽ đường tròn: CIRCLE(x,y:Integer; R:Word);
- Thủ tục ELLIPSE(x,y:integer; g1,g2,Rx,Ry:Word);
- Vẽ Ellip có tâm (x,y) bán kính ngang Rx, bán kính dọc Ry, góc bắt đầu là g1 và góc kết thúc là g2.

### 7.6. Định MODE đường vẽ

- Thủ tục SETWRITEMODE(Mode:Integer);
- Định Mode vẽ cho các đường thẳng.
- Ta có thể chọn Mode bằng các hằng:
  - CopyPut = 0;    XORPut = 1;
- Trong đó:
  - . CopyPut là Mode chèn, đường mới sẽ không xóa đường cũ.
  - . XORPut là Mode xóa, đường mới sẽ xóa đường cũ.

## XIII. TÔ MÀU CÁC HÌNH

### 8.1. Chọn kiểu tô

- Thủ tục SETFILLSTYLE(Pt,Cl:Word);
- Với:
  - Pt: Mẫu tô của hình. Có các hằng từ 0 đến 12.
    - 0: Tô bằng màu nền.
    - 1: Tô bằng màu viền.
    - 2: Tô bằng các dấu ---
    - .....
  - Cl: Màu tô của hình.

### 8.2. Vẽ hình chữ nhật có tô màu ở bên trong



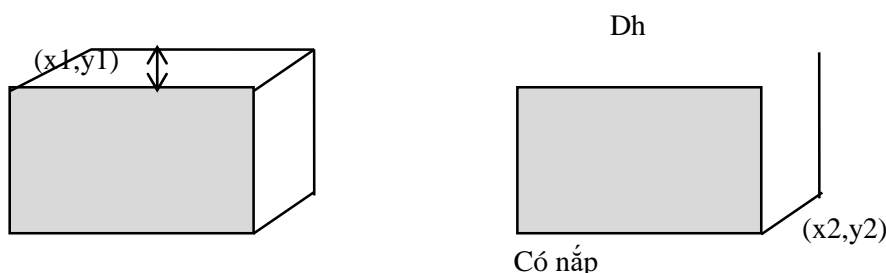
Thủ tục `BAR(x1,y1,x2,y2:Integer);`

Vẽ hình chữ nhật có tô màu và mẫu tô được xác định bởi thủ tục `SETFILLSTYLE`.

### 8.3. Vẽ hình hộp chữ nhật

Thủ tục `BAR3D(x1,y1,x2,y2,Dh:Word; Top:Boolean);`

Vẽ hình hộp chữ nhật có tọa độ đỉnh trên là  $(x1,y1)$ , đỉnh dưới là  $(x2,y2)$  và chiều dày là  $Dh$ .



Không có nắp

`Top = TRUE`: Hình hộp có nắp.

`Top = FALSE`: Hình hộp không có nắp.

### 8.4. Vẽ hình Ellip

Thủ tục `FILLELLIPSE(x,y:Integer; Rx,Ry:Word);`

### 8.5. Vẽ hình quạt tròn

Thủ tục `PIESLICE(x,y:Integer; g1,g2,R:Word);`

Vẽ hình quạt tròn có tâm  $(x,y)$ , góc đầu  $g1$ , góc cuối  $g2$ , bán kính  $R$ .

### 8.6. Vẽ hình quạt Ellip

thủ tục `SECTOR(x,y:Integer; g1,g2,Rx,Ry:Word);`

### 8.7. Làm loang màu một vùng kín

Thủ tục `FLOODFILL(x,y:Integer; Color:Word);`

Trong đó:

$(x,y)$ : điểm nằm trong vùng kín.

`Color`: màu muốn tô.

### 8.8. Vẽ đa giác

Đối với một đa giác bất kỳ có  $N$  đỉnh, ta phải khai báo  $N+1$  đỉnh để vẽ đường gấp khúc với tọa độ điểm đầu trùng với tọa độ điểm cuối.

Để vẽ đa giác ta dùng thủ tục: `DRAWPOLY(Np:Word; Var P);`

trong đó:

- `Np`: số đỉnh của đa giác + 1
- `P`: chứa tọa độ các đỉnh, là một mảng có `Np` thành phần có kiểu dữ liệu là `PointType` được định nghĩa trong Unit `Graph` như sau:

```
TYPE      PointType = Record
          x,y:Integer;
```

```
End;
```

## IX. CÁC KỸ THUẬT TẠO HÌNH CHUYỂN ĐỘNG

### 9.1. Kỹ thuật lật trang màn hình

`CARD` màn hình có nhiều trang, mỗi trang được đánh số  $0,1,2,\dots$

Để vẽ hình lên một trang màn hình, ta dùng thủ tục:

`SETACTIVEPAGE(Page:Word);`

Trong đó, `Page` là số của trang màn hình. Thủ tục này được đặt trước khi có lệnh vẽ lên màn hình.

Để đưa trang màn hình ra màn hình, ta dùng thủ tục:

`SETVISUALPAGE(Page:Word);`

`Page`: trang màn hình muốn xem.

Thông thường, màn hình sẽ làm việc và hiện ra trên trang 0. Do đó, để vừa xem màn hình vừa vẽ lên trang màn hình khác, ta thường dùng hai thủ tục trên đi kèm với nhau.

Để thực hiện tự động chương trình khi sử dụng cú pháp lật hình này, ta thường theo một giải thuật sau:

- Tạo biến `page1,page2:Word;`
- Tạo vòng lặp

...



```

Repeat
    SetVisualPage(page1); (* Xem trang màn hình page1 *)
    SetActivePage(page2); (* Vẽ hình lên trang page2 *)
    .....
    < Các thủ tục vẽ hình >
    .....
    (* Hoán vị 2 biến page1, page2 *)
    Temp:=page1;
    page1:=page2;
    page2:=Temp;
Until <Điều kiện thoát>;

```

## 9.2. Lưu và di chuyển một vùng màn hình

Chúng ta có thể lưu một vùng màn hình vào bộ nhớ rồi sau đó lại dán nó lên màn hình tại một vị trí khác.

Lưu một vùng màn hình vào bộ nhớ được thực hiện bằng thủ tục:

```
GETIMAGE(x1,y1,x2,y2:Integer; Var P:Pointer);
```

trong đó P là con trỏ để lưu nội dung của vùng (x1,y1,x2,y2).

Việc đăng ký một vùng nhớ động phải được khai báo dung lượng cần thiết. Dung lượng vùng nhớ được thực hiện bằng hàm:

```
IMAGESIZE(x1,y1,x2,y2:Integer):Word;
```

Để hiện hình ảnh từ bộ nhớ ra màn hình, ta dùng thủ tục:

```
PUTIMAGE(x,y:Integer; P:Pointer; Mode:Word);
```

trong đó:

(x,y): Tọa độ đỉnh trái hình chữ nhật mà ta muốn đưa ra.

P : Con trỏ lưu vùng hình chữ nhật.

Mode: Hằng số chỉ phương thức hiện ra màn hình. Mode chứa một trong các hằng sau:

NormalPut = 0: Xuất ra như đã lưu (phép MOV)

XORPut = 1: Phép XOR, xóa hình cũ nếu hai hình giao nhau.

ORPut = 2: Phép OR, lấy cả hai hình nếu hai hình giao nhau.

ANDPut = 3: Phép AND, nếu hai hình giao nhau thì lấy phần chung.

NOTPut = 4: Phép NOT, cho ra âm bản.

Về việc thực hiện ta tiến hành như sau:

- Khai báo một biến con trỏ P:Pointer;
- Đăng ký một vùng nhớ động do P quản lý bằng thủ tục  
GETMEM(P,ImageSize(x1,y1,x2,y2));
- Lưu hình ảnh bằng thủ tục GETIMAGE(x1,y1,x2,y2,P^);
- Xuất ra màn hình bằng thủ tục PUTIMAGE(x,y,P^,Mode);

## BÀI TẬP BỔ SUNG

Bài 1: Viết chương trình để tìm lời giải cho bài toán sau:

Trong giỏ vừa thỏ vừa gà,

Một trăm cái cẳng bốn ba cái đầu.

Hỏi có mấy gà mấy thỏ?

Bài 2: Viết chương trình để tìm lời giải cho bài toán sau:

Trăm trâu trăm bó cỏ

Bó lại cho tròn

Trâu đứng ăn năm

Trâu nằm ăn ba

Năm trâu nghé ăn một.

Hỏi có bao nhiêu trâu đứng, trâu nằm, trâu nghé?



Bài 3: Viết chương trình nhập vào các số nguyên từ bàn phím cho đến khi nào gặp số nguyên tố thì kết thúc nhập. Tính tổng các số chẵn và trung bình cộng các số lẻ.

**Gợi ý:**

Dùng vòng lặp REPEAT ... UNTIL NTo; để nhập. Trong đó, NTo là biến kiểu Boolean để kiểm tra số được nhập vào có phải là số nguyên tố hay không.

Bài 4: Viết chương trình nhập vào một số nguyên dương. Hãy thông báo lên màn hình số đó có bao nhiêu chữ số và tổng các chữ số của số đó.

**Gợi ý:**

Dùng vòng lặp WHILE. Trong khi  $N > 0$  thì: lấy ra chữ số cuối cùng của  $N$  để tính bằng phép toán MOD 10, sau đó bỏ bớt đi chữ số cuối cùng của  $N$  bằng phép toán DIV 10.

Bài 5: Viết chương trình in ra màn hình tất cả các số nguyên tố từ 2 đến N. Với N được nhập từ bàn phím.

Bài 6: Viết chương trình phân tích một số ra thừa số nguyên tố. Ví dụ: N=100 sẽ in ra màn hình:

|     |   |
|-----|---|
| 100 | 2 |
| 50  | 2 |
| 25  | 5 |
| 5   | 5 |
| 1   |   |

Bài 7: Viết chương trình in ra các số nguyên từ 1 đến  $N^2$  theo hình xoắn ốc với N được nhập vào từ bàn phím. Ví dụ, với N=5 ta có:

|    |    |    |    |   |
|----|----|----|----|---|
| 1  | 2  | 3  | 4  | 5 |
| 16 | 17 | 18 | 19 | 6 |
| 15 | 24 | 25 | 20 | 7 |
| 14 | 23 | 22 | 21 | 8 |
| 13 | 12 | 11 | 10 | 9 |

Bài 8: Viết chương trình nhập vào số tự nhiên  $N$  ( $N$  lẻ), sau đó điền các số từ 1 đến  $n^2$  vào trong một bảng vuông sao cho tổng các hàng ngang, hàng dọc và 2 đường chéo đều bằng nhau (bảng này được gọi là Ma phương).

Ví dụ: Với  $N=3$  và  $N=5$  ta có

|   |   |   |
|---|---|---|
| 2 | 7 | 6 |
| 9 | 5 | 1 |
| 4 | 3 | 8 |

Tây

| Bắc |    |    |    |    |
|-----|----|----|----|----|
| 3   | 16 | 9  | 22 | 15 |
| 20  | 8  | 21 | 14 | 2  |
| 7   | 25 | 13 | 1  | 19 |
| 24  | 12 | 5  | 18 | 6  |
| 11  | 4  | 17 | 10 | 23 |

Đông

Nam

**Phương pháp:**

Xuất phát từ ô bên phải của ô nằm giữa. Đi theo **hướng đông bắc** để điền các số 1, 2, ...

Khi điền số, cần chú ý một số nguyên tắc sau:

- Nếu vượt ra phía ngoài bên phải của bảng thì quay trở lại cột đầu tiên.
- Nếu vượt ra phía ngoài bên trên của bảng thì quay trở lại dòng cuối cùng.
- Nếu số đã điền k chia hết cho N thì số tiếp theo sẽ được viết trên cùng một hàng với k nhưng cách

1 ô về phía bên phải.

Uses Crt;

```
Var A:Array[1..20,1..20] Of Word;
```

n,i,j,k:Word;

Begin

```
Write('Nhap N= '); Readln(n);
```



```

Clrscr;
{Định vị ô xuất phát}
i:=n DIV 2 + 1;
j:=n DIV 2 + 2;

{Điền các số k từ 1 đến n*n}
For k:=1 To n*n Do
Begin
  A[i,j]:=k;
  If k MOD n=0 Then j:=j+2
  Else Begin
    {Đi theo hướng đồng bắc}
    j:=j+1; i:=i-1;
  End;
  If j>n Then j:=j MOD n;
  If i=0 Then i:=n;
End;

{In kết quả ra màn hình}
For i:=1 To n Do
Begin
  For j:=1 To n Do write(a[i,j]:4);
  Writeln;
End;
Readln;
End.

```

Bài 9: Viết chương trình nhập vào 2 mảng số nguyên A, B đại diện cho 2 tập hợp (không thể có 2 phần tử trùng nhau trong một tập hợp). Trong quá trình nhập, phải kiểm tra: nếu phần tử vừa nhập vào đã có trong mảng thì không bổ sung vào mảng. In ra màn hình các phần tử là giao của 2 tập hợp A, B.

**Ý tưởng:**

Duyệt qua tất cả các phần tử  $a_i \in A$ . Nếu  $a_i \in B$  thì viết  $a_i$  ra màn hình.

```

Uses Crt;
Type Mang=ARRAY[1..50] Of Integer;
Var A,B:Mang;
    n,m:Byte;

Function KiemTra(x:Integer; n:Byte; A:Mang):Boolean;
Var i:Byte; Found:Boolean;
Begin
  Found:=False;
  i:=1;
  While (i<=n) AND (not Found) Do
    If x=A[i] Then Found:=True Else i:=i+1;
  KiemTra:=Found;
End;

Procedure NhapMang(Var n:Byte; Var A:Mang);
Var ch:Char;
    x:Integer;
Begin
  n:=0;

```



```

Repeat
  Write('x='); Readln(x);
  If not KiemTra(x,n,A) Then
    Begin
      n:=n+1; A[n]:=x;
    End;
  Writeln('An ESC de ket thuc nhap!');
  ch:=Readkey;
Until ch=#27;
End;

Procedure GiaoAB(n:Byte; A:Mang;m:Byte; B:Mang);
Var i:Byte;
Begin
  For i:=1 To n Do
    If KiemTra(A[i],m,B) Then Write(A[i]:4);
End;

```

```

Begin
  Clrscr;
  Writeln('Nhap mang A: ');
  NhapMang(n,A);
  Writeln('Nhap mang B: ');
  NhapMang(m,B);
  Writeln('Giao cua 2 mang A&B la: ');
  GiaoAB(n,A,m,B);
  Readln;
End.

```

Bài 10: Cho một mảng số nguyên gồm  $n$  phần tử. Tìm dãy con gồm  $m$  phần tử ( $m \leq n$ ) sao cho dãy con này có tổng lớn nhất. (Dãy con là dãy các phần tử liên tiếp nhau trong mảng).

Giải:

```

Uses Crt;
Type Mang=ARRAY[1..50] Of Integer;
Var A:Mang;
    n,m,i,j,k:Byte;
    S,Max:Integer;

Begin
  Write('So phan tu cua mang: n= '); Readln(n);
  For i:=1 To n Do
    Begin
      Write('a[',i,']='); Readln(a[i]);
    End;
  Write('Nhap so phan tu cua day con: m= '); Readln(m);
  k:=1; {Vị trí phần tử đầu tiên của dãy con}
  {Giả sử m phần tử đầu tiên của mảng A là dãy con có tổng lớn nhất}
  Max:=0;
  For i:=1 To m Do Max:=Max+A[i];
  {Tìm các dãy con khác}
  For i:=2 To n-m+1 Do
    Begin
      {Tính tổng của dãy con thứ i}
      S:=0;

```



```

For j:=i To i+m-1 Do S:=S+A[j];
If S>Max Then {Nếu dãy con tìm được có tổng lớn hơn dãy con trước}
Begin
  Max:=S; {Thay tổng mới}
  k:=i;    {Thay vị trí đầu tiên của dãy con mới}
End;
End;
Writeln('Dãy con có tổng lớn nhất là:');
For i:=k To k+m-1 Do Write(A[i]:5);
Readln;
End.

```

## PHỤ LỤC

### TÀI FREE PASCAL PHỤC VỤ HỌC TẬP VÀ NGHIÊN CỨU

Bạn có thể tải Free Pascal theo các cách sau:

❖ Đối với máy tính sử dụng Hệ điều hành Windows (XP, Vista, 7, 8, 8.1):

- ☞ [http://www.mediafire.com/download/dicrwasenuq79ok/Free\\_Pascal\\_2.4.4.exe](http://www.mediafire.com/download/dicrwasenuq79ok/Free_Pascal_2.4.4.exe)
- ☞ <http://www.mediafire.com/?u27v2uo7i7tff>
- ☞ <http://www.freepascal.org/download/i386/win32-austria.var>
- ☞ Trang chủ: <http://www.freepascal.org>



❖ Đối với các thiết bị di động (Smart Phone):

☞ Hệ điều hành iOS:

- ☞ <http://vi.forios.net/tag/utilities/pascal-programming-language-download-174137.html>

☞ Hệ điều hành WindowsPhone:

- ☞ <http://www.windowsphone.com/vi-vn/store/app/pascal-l%C3%BCders/aa61aac6-ef28-42de-a524-3200b54c5891>

hoặc

- ☞ <http://www.windowsphone.com/en-us/store/app/pascal-interpreter-for-wp7/824e28f0-9e6e-e011-81d2-78e7d1fa76f8>

☞ Hệ điều hành Android:

- ☞ [https://play.google.com/store/apps/details?id=me.zed\\_0xff.android.pascal&hl=vi](https://play.google.com/store/apps/details?id=me.zed_0xff.android.pascal&hl=vi)

hoặc

- ☞ <http://choapp.vn/tai-app-android/pascalgui-pascal-compiler/2929543/1/26/>



### LIÊN LẠC

- ☞ Đóng góp ý kiến cho tài liệu này tại: <http://www.facebook.com/nguyentrongminhhongphuoc>

- ☞ Email: hongphuoc123456@gmail.com





# MỤC LỤC

## CHƯƠNG I: MỘT SỐ KHÁI NIỆM VỀ NGÔN NGỮ LẬP TRÌNH

|                                             |   |
|---------------------------------------------|---|
| Bài 1: Khái niệm .....                      | 1 |
| Bài 2: Các thành phần cơ bản của NNLT ..... | 1 |

## CHƯƠNG II: CHƯƠNG TRÌNH ĐƠN GIẢN

|                                     |    |
|-------------------------------------|----|
| Bài 1: Cấu trúc chương trình .....  | 1  |
| Bài 2: Các kiểu dữ liệu chuẩn ..... | 2  |
| Bài 3: Thủ tục chuẩn vào/ra .....   | 4  |
| Bài tập .....                       | 4  |
| Đọc thêm .....                      | 10 |

## CHƯƠNG III: CẤU TRÚC Rẽ NHÁNH VÀ LẶP

|                                          |    |
|------------------------------------------|----|
| Bài 1: Cấu trúc rẽ nhánh .....           | 11 |
| Bài 2: Cấu trúc lặp .....                | 11 |
| Bài 3: Câu lệnh lựa chọn Case...of ..... | 12 |
| Bài tập .....                            | 12 |
| Đọc thêm .....                           | 26 |

## CHƯƠNG IV: KIỂU DỮ LIỆU CÓ CẤU TRÚC

|                           |    |
|---------------------------|----|
| Bài 1: Kiểu mảng .....    | 28 |
| Bài 2: Kiểu xâu .....     | 29 |
| Bài 3: Kiểu bản ghi ..... | 29 |
| Bài tập .....             | 30 |

## CHƯƠNG V: TỆP VÀ THAO TÁC VỚI TỆP

|                               |    |
|-------------------------------|----|
| Tệp và thao tác với tệp ..... | 39 |
| Đọc thêm .....                | 41 |
| Bài tập .....                 | 41 |

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

|                                              |    |
|----------------------------------------------|----|
| Bài 1: Chương trình con và phân loại .....   | 44 |
| Bài 2: Thư viện chương trình con chuẩn ..... | 47 |
| Đọc thêm .....                               | 47 |
| Bài tập .....                                | 48 |

## MỘT SỐ BÀI TẬP TỰ LUYỆN .....

## MỘT SỐ HÀM VÀ THỦ TỤC .....

## PHẦN BỔ SUNG .....

|                       |    |
|-----------------------|----|
| Bài đọc thêm 1 .....  | 70 |
| Bài đọc thêm 2 .....  | 71 |
| Bài đọc thêm 3 .....  | 71 |
| Bài đọc thêm 4 .....  | 74 |
| Bài tập bổ sung ..... | 79 |
| Phụ lục .....         | 83 |
| Mục lục .....         | 84 |



Success is the ability to go from  
one failure to another with no  
loss of enthusiasm.

*Thành công là khả năng đi từ thất bại này đến thất bại  
khác mà không mất đi nhiệt huyết.*



