

c) Chú thích

Có thể đặt các đoạn chú thích trong chương trình nguồn. Các chú thích này giúp cho người đọc chương trình nhận biết ý nghĩa của chương trình đó dễ hơn. Chú thích không ảnh hưởng đến nội dung chương trình nguồn và được chương trình dịch bỏ qua.

Trong Pascal các đoạn chú thích được đặt giữa cặp dấu { và } hoặc (* và *). Một trong những cách tạo chú thích trong C++ là đặt chúng giữa cặp dấu /* và */.

TÓM TẮT

- Cần có chương trình dịch để chuyển chương trình nguồn thành chương trình đích.
- Có hai loại chương trình dịch: thông dịch và biên dịch.
- Các thành phần của ngôn ngữ lập trình: bảng chữ cái, cú pháp và ngữ nghĩa.
- Mọi đối tượng trong chương trình đều phải được đặt tên:
 - Tên dành riêng: Được dùng với ý nghĩa riêng, không được dùng với ý nghĩa khác.
 - Tên chuẩn: Tên dùng với ý nghĩa nhất định, khi cần dùng với ý nghĩa khác thì phải khai báo.
 - Tên do người lập trình đặt: Cần khai báo trước khi sử dụng.
- Hằng: Đại lượng có giá trị không thay đổi trong quá trình thực hiện chương trình.
- Biến: Đại lượng được đặt tên. Giá trị của biến có thể thay đổi trong quá trình thực hiện chương trình.

CÂU HỎI VÀ BÀI TẬP

1. Tại sao người ta phải xây dựng các ngôn ngữ lập trình bậc cao?
2. Chương trình dịch là gì? Tại sao cần phải có chương trình dịch?
3. Biên dịch và thông dịch khác nhau như thế nào?
4. Hãy cho biết các điểm khác nhau giữa tên dành riêng và tên chuẩn.
5. Hãy tự viết ra ba tên đúng theo quy tắc của Pascal.
6. Hãy cho biết những biểu diễn nào dưới đây không phải là biểu diễn hằng trong Pascal và chỉ rõ lỗi trong từng trường hợp:
 - a) 150.0 b) -22 c) 6,23 d) '43'
 - e) A20 f) 1.06E-15 g) 4+6 h) 'C
 - i) 'TRUE'

1. Mục đích, yêu cầu

- Giới thiệu một chương trình Pascal hoàn chỉnh đơn giản;
- Làm quen với một số dịch vụ cơ bản của Turbo Pascal hoặc Free Pascal trong việc soạn thảo, lưu trữ, dịch và thực hiện chương trình.

2. Nội dung

a) Gõ chương trình sau:

```
program Giai_PTB2;
uses crt;
var a, b, c, D: real;
    x1, x2: real;
begin
  clrscr;
  write('a, b, c: ');
  readln(a,b,c);
  D:= b*b - 4*a*c;
  x1:= (-b - sqrt(D))/(2*a);
  x2:= -b/a - x1;
  write('x1 = ', x1:6:2, ' x2 = ', x2:6:2);
  readln
end.
```

Chú ý: – Dấu chấm phẩy (;) dùng để ngăn cách các khai báo và các câu lệnh.
Có thể bỏ qua dấu chấm phẩy sau câu lệnh trước từ khoá **end**.
– Sau từ khoá **end** cuối chương trình phải đặt dấu chấm.

b) Nhấn phím **F2** và lưu chương trình với tên là **PTB2.PAS** lên đĩa.

c) Nhấn tổ hợp phím **Alt+F9** để dịch và sửa lỗi cú pháp (nếu có).

d) Nhấn tổ hợp phím **Ctrl+F9** để thực hiện chương trình. Nhập các giá trị 1; -3 và 2. Quan sát kết quả hiển thị trên màn hình ($x_1 = 1.00$ $x_2 = 2.00$). Nhấn phím **Enter** để quay lại màn hình soạn thảo.

e) Nhấn tổ hợp phím **Ctrl+F9** rồi nhập các giá trị 1; 0; -2.

Quan sát kết quả hiển thị trên màn hình ($x_1 = -1.41$ $x_2 = 1.41$).

- f) Chỉnh sửa chương trình trên để có chương trình không dùng biến trung gian D . Thực hiện chương trình đã sửa với các bộ dữ liệu trên.
- g) Sửa lại chương trình nhận được ở câu c) bằng cách thay đổi công thức tính x_2 (có hai cách để tính x_2).
- h) Thực hiện chương trình đã sửa với bộ dữ liệu 1; -5; 6. Quan sát kết quả trên màn hình ($x_1 = 2.00$ $x_2 = 3.00$).
- i) Thực hiện chương trình với bộ dữ liệu 1; 1; 1 và quan sát kết quả trên màn hình.

CÂU HỎI VÀ BÀI TẬP

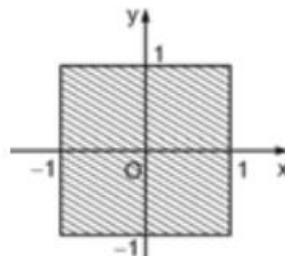
- Hãy cho biết sự khác nhau giữa hằng có đặt tên và biến.
- Tại sao phải khai báo biến?
- Trong Pascal, nếu một biến chỉ nhận giá trị nguyên trong phạm vi từ 10 đến 25532 thì biến đó có thể được khai báo bằng các kiểu dữ liệu nào?
- Biến P có thể nhận các giá trị 5; 10; 15; 20; 30; 60; 90 và biến X có thể nhận các giá trị 0,1; 0,2; 0,3; 0,4; 0,5. Khai báo nào trong các khai báo sau là đúng?

a) <code>var X,P: byte;</code>	b) <code>var P,X: real;</code>
c) <code>var P: real;</code>	d) <code>var X:real;</code>
<code>X: byte;</code>	<code>P:byte;</code>
- Để tính diện tích S của hình vuông có cạnh A với giá trị nguyên nằm trong phạm vi từ 100 đến 200, cách khai báo S nào dưới đây là đúng và tốn ít bộ nhớ nhất?

a) <code>var S: integer;</code>	b) <code>var S: real;</code>
c) <code>var S: word;</code>	d) <code>var S: longint;</code>
e) <code>var S: boolean;</code>	
- Hãy viết biểu thức toán học dưới đây trong Pascal:

$$(1+z) \frac{x + \frac{y}{z}}{1 - \frac{a}{1+x^3}}$$

- a) a/b^2 ; b) $a*b*c/2$;
c) $1/a*b/c$; d) $b/\text{sqrt}(a*a+b)$.



1. Mục đích, yêu cầu

- Xây dựng chương trình có sử dụng cấu trúc rẽ nhánh;
- Làm quen với việc hiệu chỉnh chương trình.

2. Nội dung

Bài toán. Bộ số Pi-ta-go

Biết rằng bộ ba số nguyên dương a, b, c được gọi là bộ số Pi-ta-go nếu tổng các bình phương của hai số bằng bình phương của số còn lại. Viết chương trình nhập từ bàn phím ba số nguyên dương a, b, c và kiểm tra xem chúng có là bộ số Pi-ta-go hay không.

Ý tưởng: Kiểm tra xem có đẳng thức nào trong ba đẳng thức sau đây xảy ra hay không:

$$\begin{aligned}a^2 &= b^2 + c^2 \\b^2 &= a^2 + c^2 \\c^2 &= a^2 + b^2.\end{aligned}$$

Những công việc cần thực hiện:

a) Gõ chương trình sau:

```
program Pi_ta_go;
uses crt;
var a, b, c: integer;
    a2, b2, c2: longint;
begin
  clrscr;
  write('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('Ba so da nhap la bo so Pi-ta-go')
    else writeln('Ba so da nhap khong la bo so Pi-ta-go');
readln
end.

```

Chú ý: Trước else không có dấu chấm phẩy (;).

- b) Lưu chương trình với tên **PITAGO** lên đĩa.
- c) Nhấn phím **F7** để thực hiện từng câu lệnh chương trình, nhập các giá trị $a = 3, b = 4, c = 5$.
- d) Vào bảng chọn **Debug** mở cửa sổ hiệu chỉnh để xem giá trị $a2, b2, c2$.
- e) Nhấn phím **F7** để thực hiện các câu lệnh tính những giá trị nói trên, so sánh với kết quả $a2 = 9, b2 = 16, c2 = 25$.
- f) Quan sát quá trình rẽ nhánh.
- g) Lặp lại các bước trên với bộ dữ liệu $a = 700, b = 1000, c = 800$.
- h) Nếu thay dãy lệnh

```

a2:= a;
b2:= b;
c2:= c;
a2:= a2*a;
b2:= b2*b;
c2:= c2*c;

```

bằng dãy lệnh

```

a2:= a*a;
b2:= b*b;
c2:= c*c;

```

thì kết quả có gì thay đổi với bộ dữ liệu cho ở câu g?

CÂU HỎI VÀ BÀI TẬP

1. Hãy cho biết sự giống và khác nhau của hai dạng câu lệnh **if-then**.
2. Câu lệnh ghép là gì? Tại sao phải có câu lệnh ghép?

1. Mục đích, yêu cầu

- Nâng cao kỹ năng sử dụng một số câu lệnh và một số kiểu dữ liệu thông qua việc tìm hiểu, chạy thử các chương trình có sẵn;
- Biết giải một số bài toán tính toán, tìm kiếm đơn giản trên máy tính.

2. Nội dung

Bài 1. Tạo mảng A gồm n ($n \leq 100$) số nguyên, mỗi số có trị tuyệt đối không vượt quá 300. Tính tổng các phần tử của mảng là bội số của một số nguyên dương k cho trước.

a) Hãy tìm hiểu và chạy thử chương trình sau đây:

```
program Sum1;
uses crt;
const nmax=100;
type MyArray=array[1..nmax] of integer;
var A: MyArray;
    s,n,i,k: integer;
begin
    clrscr; randomize;
    write('Nhap n =');
    readln(n);    {Tao ngau nhien mang gom n so nguyen}
    for i:= 1 to n do A[i]:= random(300)- random(300);
    for i:= 1 to n do write(A[i]:5); {in ra mang vua tao}
    writeln;
    write('Nhap k = ');
    readln(k);
    s:=0;
    for i:=1 to n do
        if A[i] mod k = 0 then s:=s+A[i];
    writeln('Tong can tinh la: ',s);
    readln
end.
```

Chú ý: Hàm chuẩn *random(n)* cho giá trị là số nguyên ngẫu nhiên trong đoạn từ 0 đến $n - 1$, còn thủ tục *randomize* khởi tạo cơ chế sinh số ngẫu nhiên.

- b) Hãy đưa các câu lệnh sau đây vào những vị trí cần thiết nhằm sửa đổi chương trình trong câu a) để có được chương trình đưa ra số các số dương và số các số âm trong mảng.

```
posi, neg: integer;
posi:= 0; neg:= 0;
if A[i]>0 then posi:= posi+1
    else if A[i]<0 then neg:= neg+1;
writeln(posi:4,neg:4);
```

Bài 2. Viết chương trình tìm phần tử có giá trị lớn nhất của 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 chỉ số nhỏ nhất.

- a) Hãy tìm hiểu chương trình sau đây:

```
program MaxElement;
const Nmax= 100;
type MyArray = array[1..Nmax] of integer;
var A: MyArray;
    n,i,j: integer;
begin
    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;
    write('Chi so: ',j, ' Gia tri: ',A[j]:4);
    readln
end.
```

- b) Chỉnh sửa chương trình trên để đưa ra chỉ số của các phần tử có cùng giá trị lớn nhất.

1. Mục đích, yêu cầu

- *Biết nhận xét, phân tích, đề xuất thuật toán giải bài toán sao cho chương trình chạy nhanh hơn;*
- *Làm quen với dữ liệu có cấu trúc và bài toán sắp xếp.*

2. Nội dung

Bài 1

- a) Hãy tìm hiểu và chạy thử chương trình thực hiện thuật toán sắp xếp dãy số nguyên bằng thuật toán tráo đổi với các giá trị khác nhau của n dưới đây. Qua đó, nhận xét về thời gian chạy của chương trình.

```
(* Chương trình giải bài toán sắp xếp dãy số *)
uses crt;
const Nmax = 250;
type ArrInt = array[1..Nmax] of integer;
var n, i, j, y: integer;
    A: ArrInt;
begin
    clrscr;
    randomize;
    write('Nhập n = ');
    readln(n);
    {Tạo ngẫu nhiên mảng gồm n số nguyên}
    for i:= 1 to n do A[i]:= random(300)-random(300);
    for i:= 1 to n do write(A[i]:5); {in mảng vừa tạo}
    writeln;
    for j:=N downto 2 do
        for i:=1 to j-1 do
```

```

        if A[i] > A[i+1] then
            begin (* Trao doi A[i] va A[i+1] *)
                t := A[i];
                A[i] := A[i+1];
                A[i+1] := t;
            end;
        writeln('Day so duoc sap xep: ');
        for i:=1 to n do
            write(A[i]: 7);
        writeln;
        readln
    end.

```

- b) Khai báo thêm biến nguyên *Dem* và bổ sung vào chương trình những câu lệnh cần thiết để biến *Dem* tính số lần thực hiện trao đổi trong thuật toán. Đưa kết quả tìm được ra màn hình.

Bài 2

Hãy đọc và tìm hiểu những phân tích để viết chương trình giải bài toán:

Cho mảng A gồm n phần tử. Hãy viết chương trình tạo mảng B[1..n], trong đó B[i] là tổng của i phần tử đầu tiên của A.

Thoạt đầu có thể viết chương trình sau để giải bài toán này:

```

program SubSum1;
const nmax=100;
type MyArray=array[1..nmax] of integer;
var A, B: MyArray;
    n, i, j: integer;
begin
    randomize;
    write('Nhap n =');
    readln(n);
    {Tao ngau nhien mang gom n so nguyen}
    for i:= 1 to n do A[i]:= random(300) - random(300);

```

```

for i:= 1 to n do write(A[i]:5);
writeln;
{Bat dau tao B}
for i:= 1 to n do
  begin
    B[i]:= 0;
    for j:= 1 to i do B[i]:= B[i]+A[j];
  end;
{Ket thuc tao B}
for i:= 1 to n do write (B[i]:6);
readln
end.

```

Để ý rằng ta có các hệ thức sau:

$$B[1] = A[1]$$

$$B[i] = B[i-1] + A[i], \quad 1 < i \leq n.$$

Do đó, ta thay đoạn chương trình từ chú thích *{Bat dau tao B}* đến *{Ket thuc tao B}* bởi hai lệnh sau:

```

B[1] := A[1];
for i:= 2 to n do B[i] := B[i-1] + A[i];

```

Với hai lệnh này, máy chỉ phải thực hiện $n - 1$ phép cộng, trong khi với đoạn chương trình trên máy phải thực hiện $\frac{n(n+1)}{2}$ phép cộng.

Nhờ việc phân tích như trên ta có thể giảm bớt đáng kể số phép toán cần thực hiện.

Tuy tốc độ tính toán của máy tính nhanh nhưng có giới hạn. Do đó, trong khi viết chương trình, ta nên tìm cách viết sao cho chương trình thực hiện càng ít phép toán càng tốt.

1. Mục đích, yêu cầu

Làm quen với việc tìm kiếm, thay thế và biến đổi chuỗi.

2. Nội dung

Bài 1. Nhập vào từ bàn phím một chuỗi. Kiểm tra chuỗi đó có phải là chuỗi đối xứng hay không. Chuỗi đối xứng có tính chất: đọc nó từ phải sang trái cũng thu được kết quả giống như đọc từ trái sang phải (còn được gọi là chuỗi palindrome).

a) Hãy chạy thử chương trình sau:

```
var i, x: byte;
    a, p: string;
begin
    write('Nhập vào chuỗi: ');
    readln(a);
    x:=length(a);    {xác định độ dài của chuỗi}
    p:='';           {khởi tạo chuỗi rỗng}
    for i:=x downto 1 do
        p:=p+a[i];    {tạo chuỗi đảo ngược}
    if a=p then
        write('Chuỗi là palindrome')
    else
        write('Chuỗi không là palindrome');
    readln
end.
```

b) Hãy viết lại chương trình trên, trong đó không dùng biến chuỗi *p*.

Bài 2. Viết chương trình nhập từ bàn phím một chuỗi ký tự *S* và thông báo ra màn hình số lần xuất hiện của mỗi chữ cái tiếng Anh trong *S* (không phân biệt chữ hoa hay chữ thường).

Bài 3. Nhập vào từ bàn phím một chuỗi. Thay thế tất cả các cụm ký tự 'anh' bằng cụm ký tự 'em'.

CÂU HỎI VÀ BÀI TẬP

1. Tại sao mảng là kiểu dữ liệu có cấu trúc?
2. Tại sao phải khai báo kích thước của mảng?
3. Các phần tử của mảng có thể có những kiểu gì?
4. Tham chiếu đến phần tử của mảng bằng cách nào?
5. Viết chương trình nhập từ bàn phím số nguyên dương N ($N \leq 100$) và dãy A gồm N số nguyên A_1, A_2, \dots, A_N có giá trị tuyệt đối không lớn hơn 1000. Hãy cho biết dãy A có phải là một cấp số cộng hay không và thông báo kết quả ra màn hình.
6. Viết chương trình nhập từ bàn phím số nguyên dương N ($N \leq 100$) và dãy A gồm N số nguyên A_1, A_2, \dots, A_N có giá trị tuyệt đối không lớn hơn 1000. Hãy đưa ra những thông tin sau:
 - a) Số lượng số chẵn và số lẻ trong dãy;
 - b) Số lượng số nguyên tố trong dãy.
7. Dãy F là dãy *Phi-bô-na-xi* nếu:
 $F_0 = 0; F_1 = 1; F_N = F_{N-1} + F_{N-2}$ với $N \geq 2$.
Viết chương trình nhập từ bàn phím số nguyên dương N và đưa ra màn hình số hạng thứ N của dãy *Phi-bô-na-xi*. Chương trình của bạn thực hiện được với giá trị lớn nhất của N là bao nhiêu?
8. Chương trình sau đây thực hiện những gì?

```
program BT8;  
const NMax = 50;  
type Mass = array[1..NMax,0..NMax-1] of real;  
var A: Mass;  
    i, j, N: byte; C: real;  
begin  
    write('Nhap N = '); readln(N);  
    for i:= 1 to N do  
        for j:= 0 to N-1 do  
            begin  
                write('A[' , i, ', ' , j, ']= '); readln(A[i, j]);  
            end;  
    for i:= 1 to N do  
        for j:= 0 to N-1 do  
            begin  
                C:= A[i, j];  
                A[i, j]:= A[N-i+1, j];  
                A[N-i+1, j]:= C;  
            end;  
end;
```

```

    for i:= 1 to N do
    begin
        for j:= 0 to N-1 do write(A[i,j]:5:2, ' ');
        writeln
    end;
    readln
end.

```

9. Cho mảng hai chiều kích thước $n \times n$ với các phần tử là những số nguyên. Tìm trong mỗi hàng phần tử lớn nhất rồi đổi chỗ nó với phần tử có chỉ số hàng bằng chỉ số cột.

Chương trình sau đây giải bài toán trên:

```

program Diag;
var
    N, i, j, Max, Ind, Vsp: integer;
    A: array[1..15, 1..15] of integer;
begin
    write('Nhap N nho hon 15: '); readln(N);
    for i:= 1 to N do
        for j:= 1 to N do
            begin
                write('A[' , i, ', ', j, ']= '); readln(A[i,j]);
            end;
        for i:= 1 to N do
            begin
                Max:= A[i,1]; Ind:= 1;
                for j:= 2 to N do
                    if A[i,j] > Max then
                        begin
                            Max:= A[i,j]; Ind:= j
                        end;
                Vsp:= A[i,i]; A[i,i]:= Max; A[i,Ind]:= Vsp;
            end;
        for i:= 1 to N do
            begin
                writeln;
                for j:= 1 to N do write(A[i,j]: 3);
            end;
        writeln
    end.

```

Hãy sửa lại chương trình trên khi thay yêu cầu tìm kiếm trong mỗi hàng bằng tìm kiếm trong mỗi cột.

10. Viết chương trình nhập từ bàn phím xâu kí tự S có độ dài không quá 100. Hãy cho biết có bao nhiêu chữ số xuất hiện trong xâu S . Thông báo kết quả ra màn hình.
11. Hãy bổ sung thêm vào chương trình *Xep_loai* (ở §13) những lệnh cần thiết để chương trình đưa ra danh sách học sinh xếp loại A.


```

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.

```

TÓM TẮT

- Việc trao đổi dữ liệu với bộ nhớ ngoài được thực hiện thông qua kiểu dữ liệu tệp.
- Để có thể làm việc với tệp cần phải khai báo biến tệp.
- Mỗi ngôn ngữ lập trình đều có các hàm/thủ tục chuẩn để làm việc với tệp.
- Các thao tác với tệp văn bản:
 - Khai báo biến tệp, mở tệp và đóng tệp.
 - Đọc/ghi: tương tự như làm việc với bàn phím và màn hình.

CÂU HỎI VÀ BÀI TẬP

1. Nêu một số trường hợp cần phải dùng tệp.
2. Trong sơ đồ thao tác với tệp, khi cần ghi dữ liệu vào tệp phải dùng những thao tác nào?
3. Tại sao cần phải có câu lệnh mở tệp trước khi đọc/ghi tệp?
4. Tại sao phải dùng câu lệnh đóng tệp sau khi đã kết thúc ghi dữ liệu vào tệp?

1. Mục đích, yêu cầu

- Rèn luyện các thao tác xử lý xâu, kỹ năng tạo hiệu ứng chữ chạy trên màn hình;
- Nâng cao kỹ năng viết, sử dụng chương trình con.

2. Nội dung

a) Trước hết, hãy tìm hiểu việc xây dựng hai thủ tục sau đây:

- Thủ tục *CatDan*(*s1*, *s2*) nhận đầu vào là xâu *s1* gồm không quá 79 ký tự, tạo xâu *s2* thu được từ xâu *s1* bằng việc chuyển ký tự đầu tiên của nó xuống vị trí cuối cùng. Ví dụ nếu *s1* = 'abcd' thì *s2* = 'bcda'.

```
type str79 = string[79];
procedure CatDan(s1: str79; var s2: str79);
begin
    s2:= copy(s1,2,length(s1)-1)+s1[1];
end;
```

- Thủ tục *CanGiua*(*s*) nhận đầu vào là xâu *s* gồm không quá 79 ký tự, bổ sung vào đầu *s* một số dấu cách để khi đưa ra màn hình xâu ký tự *s* ban đầu được căn giữa dòng (80 ký tự).

```
procedure CanGiua(var s: str79);
var i, n: integer;
begin
    n:= length(s);
    n:= (80-n) div 2;
    for i:= 1 to n do s:= ' '+s;
end;
```

b) Theo dõi cách sử dụng hai thủ tục trên, ta có thể viết chương trình sau đây để nhập một xâu ký tự từ bàn phím và đưa xâu đó ra màn hình có dạng dòng chữ chạy giữa màn hình văn bản 25×80.

```
uses crt;
type str79 = string[79];
```

```

var s1, s2: str79;
    stop: boolean;
procedure CatDan(s1: str79; var s2: str79);
begin
    s2:= copy(s1,2,length(s1)-1)+s1[1];
end;
procedure CanGiua(var s: str79);
var i, n: integer;
begin
    n:= length(s);
    n:= (80-n) div 2;
    for i:= 1 to n do s:= ' '+s;
end;
begin
    clrscr;
    write('Nhap xau s1: '); readln(s1);
    CanGiua(s1);
    clrscr;
    stop:= false;
    while not(stop) do
        begin
            gotoxy(1,12); (* Chuyen con tro den dau dong 12*)
            write(s1);
            delay(500); (* Dung 500 miligiay *)
            CatDan(s1, s2);
            s1:=s2;
            stop:=keypressed; (* Nhan mot phim bat ki de ket thuc*)
        end;
    readln
end.

```

Hãy chạy thử chương trình trên với dòng chữ

'... Mừng nghìn năm Thăng Long - Hà Nội!... '

- c) Hãy viết thủ tục *ChuChay(s, dong)* nhận đầu vào là xâu *s* gồm không quá 79 kí tự và biến nguyên *dong*, đưa ra xâu *s* có dạng chữ chạy ở dòng *dong*. Viết và chạy chương trình có sử dụng thủ tục này.

1. Mục đích, yêu cầu

- Nâng cao kỹ năng viết, sử dụng chương trình con;
- Biết cách viết một chương trình có cấu trúc để giải một bài toán trên máy tính.

2. Nội dung

- a) Tìm hiểu việc xây dựng các hàm và thủ tục thực hiện tính độ dài các cạnh, chu vi, diện tích, kiểm tra các tính chất đều, cân, vuông của tam giác được trình bày dưới đây.

Giả thiết tam giác được xác định bởi toạ độ của ba đỉnh. Ta sử dụng kiểu bản ghi để mô tả một tam giác:

```
type Diem = record
    x, y: real;
end;
Tamgiac = record
    A, B, C: Diem;
end;
```

Ta xây dựng các thủ tục và hàm:

- Thủ tục nhận dữ liệu vào là biến mô tả tam giác R và dữ liệu ra là độ dài của ba cạnh a, b, c :
`procedure Daicanh(var R: Tamgiac; var a, b, c: real);`
- Hàm tính chu vi của tam giác R :
`function Chuvi(var R: Tamgiac): real;`
- Hàm tính diện tích của tam giác R :
`function Dientich(var R: Tamgiac): real;`
- Thủ tục nhận đầu vào là biến mô tả tam giác R và đầu ra là tính chất của tam giác (*Deu* hay *Can* hay *Vuong*):
`procedure Tinhchat(var R: Tamgiac; var Deu, Can, Vuong: boolean);`

- Thủ tục hiển thị tọa độ ba đỉnh tam giác lên màn hình:

```
procedure Hienthi(var R: Tamgiac);
```

- Hàm tính khoảng cách giữa hai điểm P, Q :

```
function Kh_cach(P,Q: Diem): real;
```

- b) Tìm hiểu chương trình nhập vào tọa độ ba đỉnh một tam giác và sử dụng các hàm, thủ tục được xây dựng dưới đây để khảo sát các tính chất của tam giác.

```
uses crt;
const eps = 1.0E-6;
type
    Diem = record
        x, y: real;
    end;
    Tamgiac = record
        A, B, C: Diem;
    end;
var T: Tamgiac;
    Deu, Can, Vuong: boolean;
function Kh_cach(P, Q: Diem): real;
begin
    Kh_cach:=sqrt((P.x-Q.x)*(P.x-Q.x)+(P.y-Q.y)*(P.y-Q.y));
end;
procedure Daicanh(var R: Tamgiac; var a, b, c: real);
begin
    a:= Kh_cach(R.B, R.C);
    b:= Kh_cach(R.A, R.C);
    c:= Kh_cach(R.A, R.B);
end;
function ChuVi(var R: Tamgiac): real;
var a, b, c: real;
begin
    Daicanh(R, a, b, c);
    ChuVi:= a + b + c;
end;
```

```

function Dientich(var R: Tamgiac): real;
    var a, b, c, p: real;
    begin
        Daicanh(R, a, b, c);
        p:= (a+b+c)/2;
        Dientich:= sqrt(p*(p-a)*(p-b)*(p-c))
    end;

procedure Hienthi(var R: tamgiac);
    begin
        writeln('Toa do 3 dinh cua tam giac la: ');
        writeln(' - Dinh A(',R.A.x:0:3,', ', ' R.A.y:0:3,')');
        writeln(' - Dinh B(',R.B.x:0:3,', ', ' R.B.y:0:3,')');
        writeln(' - Dinh C(',R.C.x:0:3,', ', ' R.C.y:0:3,')');
    end;

procedure Tinhchat(var R:Tamgiac;var Deu,Can,Vuong:boolean);
    var a, b, c: real;
    begin
        Deu:= false; Can:= false; Vuong:= false;
        Daicanh(R, a, b, c);
        if (abs(a-b)<eps) and (abs(a-c)<eps) then
            Deu:= true
        else
            if (abs(a-b)<eps) or (abs(a-c)<eps) or (abs(b-c)<eps)
            then Can:= true;
            if (abs(a*a+b*b-c*c)<eps) or (abs(a*a+c*c-b*b)<eps)
            or (abs(b*b+c*c-a*a)<eps) then Vuong:= true;
        end;

begin
    writeln('Nhap tam giac: ');
    write('Toa do dinh A: '); readln(T.A.x, T.A.y);
    write('Toa do dinh B: '); readln(T.B.x, T.B.y);
    write('Toa do dinh C: '); readln(T.C.x, T.C.y);

```



```

writeln('=====');
Hienthi(T);
writeln('Dien tich: ',Dientich(T):9:3);
writeln('Chu vi: ',Chuvi(T):9:3);
Tinhchat(T, Deu, Can, Vuong);
writeln('Tam giac co tinh chat: ');
if Deu then writeln(' la tam giac deu')
    else if Can then writeln(' la tam giac can');
if Vuong then writeln(' la tam giac vuong');
readln;
end.

```

- c) Viết chương trình sử dụng các hàm và thủ tục xây dựng ở trên để giải bài toán:

Cho tệp dữ liệu TAMGIAC.DAT có cấu trúc như sau:

- Dòng đầu tiên chứa số N ;
- N dòng tiếp theo, mỗi dòng chứa sáu số thực $x_A, y_A, x_B, y_B, x_C, y_C$ là tọa độ ba đỉnh $A(x_A, y_A), B(x_B, y_B), C(x_C, y_C)$ của tam giác ABC .

Hãy nhập dữ liệu từ tệp đã cho và trong số N tam giác đó, đưa ra tệp TAMGIAC.OUT gồm ba dòng:

- Dòng đầu tiên là số lượng tam giác đều;
- Dòng thứ hai là số lượng tam giác cân (nhưng không là đều);
- Dòng thứ ba là số lượng tam giác vuông.