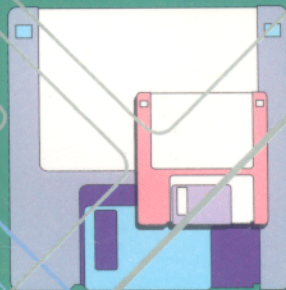


BÙI VIỆT HÀ
(CHỦ BIÊN)

TỰ HỌC LẬP TRÌNH

PASCAL

TẬP 4



School net



NHÀ XUẤT BẢN GIÁO DỤC

BÙI VIỆT HÀ
NGUYỄN QUANG HÒA

TỰ HỌC LẬP TRÌNH PASCAL

Tập 4

(Có thể tham khảo để dạy cho học sinh lớp 9)

NHÀ XUẤT BẢN GIÁO DỤC – 1999

Phần 1. BÀI TẬP

§1 ÔN TẬP

DỮ LIỆU VÀ THUẬT TOÁN TRONG PASCAL

Câu hỏi, ví dụ, bài tập

1. Hai kiểu khai báo sau có tương đương với nhau không:

Var Ch: Char;

Var St: String[1];

Sau đó liệu ta có thể thực hiện các lệnh gán như sau hay không:

a. St:=Ch;

b. Ch:=St;

2. Trong chương trình Pascal nếu ta khai báo một biến số nhưng không đặt giá trị ban đầu cho chúng thì các biến này sẽ như thế nào. Để minh họa xét ví dụ sau:

Uses Cr;

Var n: integer;

St: String;

a: Real;

Begin

Writeln ('Số tự nhiên:',n);

Writeln ('Xâu kí tự:',St);

Writeln ('Số thực:',a:10:2);

End.

3. Lập chương trình in ra trên màn hình 10 dòng đầu tiên của tam giác Pascal theo dạng sau:

```

          1
        1   1
      1   2   1
    1   3   3   1
  1   4   6   4   1
... ..

```

4. Lập chương trình in ra trên màn hình 10 dòng đầu tiên của tam giác Pascal theo dạng sau:

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
.....

```

5. Giả sử giá trị ban đầu của X là 0.1. Hãy tính giá trị của X sau các lệnh sau:

- If $2 + 4 = 5$ then $X := X + 1$;
- If $(1 + 2 = 3)$ or $(4 < 1)$ then $X := \text{round}(X + 1)$;
- If $((2 > 3)$ or not $((1 + 1 = 2)$ and $(3 < 5)))$ then $X := X * X$;

6. Hãy kiểm tra kết quả của chương trình sau:

```

Program VD;
Uses Crt;
Const Nmax = 1000;
Type DayNP = Array[0..Nmax] of byte;
Var A: DayNP;

```

```

N,i:integer;
Begin
  Readln(N);
  For i:=0 to N do A[i]:=0;
  While A[N] <> 1 do
    Begin
      For i:=1 to N do Write(A[N-i]);
      Writeln;
      i:=0;
      While A[i] = 1 do
        Begin
          A[i]:=0; i:=i+1;
        End;
      A[i]:=1;
    End;
  End.

```

7. Nhập vào ký tự có dạng sau từ bàn phím:

$$X_1 \pm X_2 \pm X_3 \pm \dots \pm X_k$$

có dạng: các chữ số nối tiếp nhau bằng dấu + hoặc -.Viết chương trình tính giá trị của biểu thức trên.

8. Có một chuyên gia lập trình Pascal phát biểu là bất cứ một lệnh lặp While nào cũng có thể thay thế tương đương bằng một lệnh lặp Repeat. Ngược lại mọi lệnh Repeat đều có thể thay thế bằng lệnh While.

Điều đó có đúng không? Hãy cho ví dụ minh họa.

9. Một lệnh lặp For có thể thay thế bằng lệnh While không. Điều ngược lại có đúng không?

10. Trong Pascal có lệnh tạo nhãn (Label) và lệnh chuyển điều khiển đến nhãn (Goto). Trên thực tế có thể thấy rằng lệnh tạo nhãn và chuyển điều khiển tới nhãn dễ gây rối chương trình và khó theo dõi. Em hãy chuyển chương trình sau có nhãn thành không có nhãn mà tác dụng của chương trình không thay đổi.

```
Program VD;  
Label 100, 101;  
Var i, a: integer;  
Begin  
  For i:=1 to 100 do  
    Begin  
      Writeln('Nhập a: '); Readln(a);  
      If a <= 0 then goto 100;  
      If a mod 5 = 0 then  
        Begin  
          Writeln(a); goto 101;  
        End;  
    100: End;  
    Writeln('Không tìm thấy');  
  101: End.
```



Bài tập về nhà

11. Viết chương trình nhập số tự nhiên N và in ra dòng thứ N của tam giác Pascal.

12. Xét bàn cờ vua 8×8 . Các ô được đánh dấu bởi vị trí hàng và cột, ví dụ vị trí (1,2) chỉ ra vị trí hàng 1 cột 2.

Cho trước k vị trí, trên đó đã xếp k quân hậu: $(i_1, j_1), (i_2, j_2), \dots, (i_k, j_k)$.

a. Viết chương trình kiểm tra xem k quân hậu trên có ở trong trạng thái độc lập hay không (tức là không quân nào có thể khống chế quân nào).

b. Trường hợp các quân trên đã độc lập hãy kiểm tra xem có thể bổ xung thêm một quân hậu nữa mà vẫn bảo đảm tính độc lập được hay không, nếu được hãy chỉ ra vị trí của quân hậu mới.

13. Xét bàn cờ vua 8×8 . Các ô được đánh dấu bởi vị trí hàng và cột, ví dụ vị trí $(1,2)$ chỉ ra vị trí hàng 1 cột 2. Cho trước k vị trí, trên đó đã xếp k quân tốt: $(i_1, j_1), (i_2, j_2), \dots, (i_k, j_k)$.

Hãy viết chương trình kiểm tra xem có thể xếp một quân hậu trên bàn cờ sao cho nó có thể khống chế toàn bộ k quân tốt. Nếu được hãy chỉ ra vị trí của quân hậu đó.

14. Xét bàn cờ vua 8×8 . Các ô được đánh dấu bởi vị trí hàng và cột, ví dụ vị trí $(1,2)$ chỉ ra vị trí hàng 1 cột 2. Cho trước k vị trí, trên đó đã xếp k quân tốt: $(i_1, j_1), (i_2, j_2), \dots, (i_k, j_k)$.

Hãy viết chương trình để chỉ ra và xếp thêm trên bàn cờ một số tối thiểu nhất các con hậu sao cho chúng sẽ khống chế toàn bộ k quân tốt. Chương trình phải chỉ ra được vị trí của mỗi quân hậu được xếp.

15. Trên một bàn tròn người ta xếp n số 0 và 1. Giả sử dãy số đã cho được viết dưới dạng: $a_1 a_2 a_3 \dots a_n$. Sau mỗi phút các số trên vòng tròn sẽ tự biến đổi theo cách sau: Nếu hai số bên cạnh giống nhau thì không thay đổi, ngược

lại nếu hai số bên cạnh khác nhau thì sẽ thay đổi $0 \rightarrow 1$ và $1 \rightarrow 0$.

Cho trước dãy số như trên, lập chương trình tính sau k phút (k nhập từ bàn phím) dãy số ban đầu sẽ thay đổi như thế nào.

16. Giả thiết giống bài trên. Để thấy rằng dãy đã cho sẽ không biến đổi nữa khi mà tất cả các số của dãy là như nhau.

Cho trước một dãy bất kỳ ban đầu, hãy kiểm tra xem quá trình biến đổi trên có bị dừng lại hay không. Hãy kiểm tra với các dãy cụ thể sau:

a. 101010101010

b. 100100100100

17. Giả thiết giống bài trên. Dãy đã cho ban đầu được gọi là Tuần hoàn nếu quá trình biến đổi đến một lúc nào đó sẽ xuất hiện lại dãy ban đầu.

Cho trước một dãy ban đầu hãy kiểm tra xem nó có phải là tuần hoàn hay không. Hãy kiểm tra cụ thể với các dãy sau:

a. 1010101000101100

b. 0100010101111001

18. Cần phải xếp 20 bao gạo vào 8 nhà kho sao cho mỗi cạnh của nhà kho phải có tổng số bao gạo là 7. Người ta đã nghĩ ra cách xếp như sau:

2	3	2
3		3
2	3	2

a. Giải quyết bài toán trên nhưng với tổng số 24 bao gạo.

b. Giải quyết bài toán chỉ với 16 bao gạo.

19. Ma trận kỳ ảo bậc N là bảng số vuông $N \times N$ bao gồm các số $1, 2, 3, \dots, N^2$ sao cho tổng số các hàng, các cột và các đường chéo đều bằng nhau.

Em hãy thử tìm ma trận kỳ ảo bậc 4.

20. Hãy thử tìm ma trận kỳ ảo bậc 5.

§2. KHÁI NIỆM THỦ TỤC



Câu hỏi, ví dụ, bài tập

1. Trong phần khai báo của thủ tục có thể khai báo một thủ tục khác được hay không? Cho ví dụ.

2. Trong chương trình của một thủ tục có thể sử dụng các biến:

a. Được khai báo trong chương trình chính hay không?

b. Được khai báo trong các thủ tục khác hay không?

3. Hãy lập một thủ tục nhập 2 số nguyên A, B ; sau đó tính và in ra màn hình thương A/B .

4. Viết thủ tục nhập dữ liệu cho một mảng số cho trước.

5. Hãy lập thủ tục tìm ma trận chuyển vị của ma trận A kích thước $m \times n$ cho trước.

6. Dùng thủ tục, đối với một ma trận $A[15 \times 15]$ tính:

a. Tổng bình phương các số hạng nằm trên đường chéo chính.

- b. Tổng lập phương các số hạng nằm trên hàng lẻ.
7. Dùng thủ tục, đối với một mảng các số thực, hãy tính tổng của các số hạng âm và in kết quả ra màn hình.
8. Viết một thủ tục nhập một mảng hai chiều từ bàn phím.
9. Dùng thủ tục chuyển một số tự nhiên n cho trước sang hệ cơ số 2.
10. Dùng thủ tục giải phương trình bậc nhất: $ax+b=0$.
11. Dùng thủ tục giải phương trình bậc hai: $ax^2+bx+c=0$.



Bài tập về nhà

12. Có thể khai báo tên của thủ tục trùng với tên của một thủ tục có sẵn trong Pascal được không?

13. Dùng thủ tục giải hệ phương trình tuyến tính 2 ẩn cho trước (các hệ số được nhập vào từ bàn phím):

$$\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2 \end{cases}$$

14. Hãy viết lại thủ tục INSERT đối với một chuỗi ký tự cho trước tùy ý.

15. Hãy viết lại thủ tục DELETE đối với một chuỗi ký tự cho trước tùy ý.

16. Viết một chương trình Pascal bằng cách thực hiện lần lượt các công việc sau:

- a. Lập thủ tục nhập ba số thực dương a, b, c từ bàn phím.
- b. Lập thủ tục kiểm tra xem ba số trên có lập thành ba cạnh của tam giác hay không.

- c. Viết thủ tục tính diện tích tam giác.
- d. Viết thủ tục tính các trung tuyến của tam giác.
- e. Viết hoàn thiện chương trình chính.

17. Cho trước mảng $P[1..50]$ of integer, hãy viết thủ tục sinh ra mảng trên bao gồm các số nguyên tố theo thứ tự của chỉ số.

18. Viết chương trình Pascal hoàn chỉnh để thực hiện theo thực đơn như sau:

- 1- Nhập dữ liệu (nhập 3 số thực a, b, c).
- 2- Giải phương trình $ax^2 + bx + c = 0$.
- 3- Thoát khỏi chương trình.

19. Viết chương trình Pascal thực hiện theo thực đơn sau:

- 1- Nhập dữ liệu (nhập 4 số thực a, b, c, d).
- 2- Tính phần tử Min.
- 3- Tính phần tử Max.
- 4- Thoát khỏi chương trình.

20. Viết chương trình Pascal hoàn chỉnh thực hiện các công việc của thực đơn sau:

- 1- Nhập dữ liệu (nhập số tự nhiên n).
- 2- Phân tích ra thừa số nguyên tố (phân tích n thành tích các số nguyên tố).
- 3- Thoát khỏi chương trình.

21. Viết chương trình Pascal hoàn chỉnh thực hiện công việc của thực đơn:

- 1- Nhập số tự nhiên N.
- 2- Nhập cơ số q.

3- Biểu diễn N theo cơ số q.

4- Thoát.

Chú ý: Ta gọi biểu diễn số tự nhiên N theo cơ số q là biểu diễn:

$$N = a_k a_{k-1} \dots a_1 a_0 \text{ với}$$

$$N = a_k q^k + a_{k-1} q^{k-1} + \dots + a_1 q + a_0.$$

§3. THAM SỐ CỦA THỦ TỤC



Câu hỏi, ví dụ, bài tập

1. Chương trình sau đây minh họa việc truyền biến cho chương trình con. Procedure tru không có danh sách biến, không có khai báo VAR, vì vậy biến x của nó là biến của chương trình chính. Sau khi chạy xong tru biến x bị thay đổi giá trị:

```

Program VD;
Var x: integer;
(*=====*)
Procedure tru;
Begin
    x:= x-1;
    Writeln ('x-1=', x);
End;
(*=====*)
Begin {Main Program}
    Write('Hãy nhập x:'); Readln(x);
    Tru;

```

```
Writeln('Giá trị x lúc này bằng:', x);  
Readln;  
End.
```

2. Chương trình sau đây minh họa việc truyền biến cho chương trình con. Procedure *cong* không có danh sách biến nhưng có khai báo biến *x* trong phần khai báo VAR, vì vậy biến *x* của nó là biến mới (biến *x* của chương trình chính bị che). Khi chạy *cong* biến *x* của nó sẽ nhận giá trị khởi đầu bằng 0 vì chưa gán giá trị khởi động.

```
Program VD;  
Var x: integer;  
(*=====*)  
Procedure cong;  
Var x: integer;  
Begin  
    x:= x+2;  
    Writeln ('x+2 =', x)  
End;  
(*=====*)  
Begin {chương trình chính}  
    Write ('Hã y nhập x: '); Readln(x);  
    cong;  
    Writeln ('Giá trị của x lúc này bằng ', x);  
End.
```

3. Chương trình con sau đây minh họa việc truyền biến cho chương trình con.

Procedure *cong* có danh sách biến, không có khai báo VAR vì vậy biến *x* (tham trị) của nó được gán giá trị khởi

động tùy thuộc lời gọi của chương trình chính. Khi chạy *cong* biến của chương trình chính bị che vì vậy khi *cong* kết thúc biến *x* của chương trình chính vẫn mang giá trị cũ.

```
Program VD;  
Var x: integer;  
(*=====*)  
Procedure cong(x: integer);  
Begin  
    x:=x+1;  
    Writeln('x+1 =', x);  
End;  
(*=====*)  
Begin {chương trình chính}  
    Write('Hãy nhập x: '); Readln(x);  
    cong(x);  
    Writeln('Giá trị của x lúc này bằng ', x);  
End.
```

4. Bạn hãy chạy thử chương trình sau đây và giải thích xem tại sao lại như vậy:

```
Program VD;  
Var a, b: integer;  
(*=====*)  
Procedure nhan(Var a: integer);  
Begin  
    a:=a*2;  
    Writeln('a nhân 2 = ', a);  
End;
```

```
(*=====*)
Begin {chương trình chính}
    Write('Hãy nhập a: '); Readln(a);
    Write('Hãy nhập b: '); Readln(b);
    nhan (b);
    Writeln('Giá trị của a lúc này bằng ', a);
    Writeln ('Giá trị của b lúc này bằng ', b);
    Readln;
End.
```

5. Chương trình sau đây minh họa việc chương trình con sẽ khởi động các biến có trong danh sách biến theo thứ tự danh sách biến mà chương trình chính ấn định cho nó.

```
Program VD;
Var a, b: real;
(*=====*)
Procedure Enter (Var a,b: real);
Begin
    Write ('Cho biết tử số: '); Readln(a);
    Write('Cho biết mẫu số: '); Readln(b);
End;
(*=====*)
Procedure Chia(a, b: real);
Var c: real;
Begin
    c:=a/b;
    Writeln (a:5:1, '/', b:5:1, '=', c:5:1)
End;
```

```
(*=====*)
Begin {chương trình chính}
  Enter (a,b);
  Chia (b,a);
End.
```

6. Giải phương trình trong phạm vi số nguyên không âm:

$$x + y + z = 10.$$

7. Giải phương trình: $x+y+z=12$ trong phạm vi số nguyên không âm với điều kiện $x < 4$.

8. Giải phương trình: $x+y+z=13$ trong phạm vi số nguyên không âm với điều kiện $x < 4$, $y < 5$ và $z < 6$.

9. Cho trước các số tự nhiên N , a , b , c . Giải phương trình sau trong phạm vi số nguyên không âm: $x+y+z=N$ với điều kiện $x < a$, $y < b$ và $z < c$.



Bài tập về nhà

10. Dãy số thực được định nghĩa như sau:

$$a_1 = 2$$

$$a_{k+1} = a_k(6 - a_k^2)/4$$

Viết chương trình tính dãy trên với n khá lớn (khoảng 10000) và với độ chính xác khoảng 20 chữ số sau dấu phẩy thập phân.

Em có nhận xét gì về dãy số này?

11. Một xâu nhị phân được gọi là không lặp bậc k nếu xâu trên không chứa hai xâu con độ dài k giống nhau. Viết chương trình nhập xâu nhị phân bất kỳ từ bàn phím và kiểm tra xem nó có phải là không lặp bậc k hay không.

12. Cho trước xâu nhị phân. Hãy tính số bậc không lặp của xâu trên (tức là số lớn nhất k thỏa mã n xâu là không lặp bậc k và không là không lặp bậc $k+1$).

13. Một xâu nhị phân gọi là "chuẩn" nếu nó không chứa hai số 0 liên nhau.

a. Liệt kê và tính số các xâu nhị phân "chuẩn" độ dài 3.

b. Liệt kê và tính số các xâu nhị phân "chuẩn" độ dài 4.

c. Liệt kê và tính số các xâu nhị phân "chuẩn" độ dài 5.

d. Gọi A_n là số các xâu nhị phân chuẩn độ dài n . Hãy tính A_n .

14. Giả thiết giống như bài trên. Em hãy chứng minh công thức sau: $A_n = A_{n-1} + A_{n-2}$.

15. Viết chương trình liệt kê và tính số tất cả các xâu nhị phân độ dài 6 và không chứa 3 số 0 liên tiếp.

16. Gọi A_n là số các xâu ký tự độ dài n và chứa 3 số 0 liên tiếp. Gọi B_n là số các xâu ký tự độ dài n và không chứa 3 số 0 liên tiếp. Rõ ràng ta có $A_n + B_n = 2^n$. Tìm công thức tính A_n và B_n .

17. Cấu trúc dữ liệu Ngăn xếp:

Một mảng dữ liệu ngăn xếp là mảng dữ liệu trong đó các phần tử được lưu trữ và xử lý theo nguyên tắc "Vào sau ra trước", tức là phần tử đưa vào lần cuối cùng sẽ được đưa ra đầu tiên. (Giống như ngăn xếp, đưa vào đầu tiên nằm dưới đáy thì không thể lấy ra trước được). Đối với dữ liệu kiểu ngăn xếp (tiếng Anh gọi là Stack) chỉ có hai phép toán: Đưa vào (Push) và Lấy ra (Pull).

Hãy mô tả một kiểu dữ liệu Ngăn xếp và mô tả hai thủ tục Push và Pull.

18. Cấu trúc dữ liệu Hàng đợi:

Một mảng dữ liệu Hàng đợi là mảng dữ liệu trong đó các phần tử được lưu trữ và xử lý theo nguyên tắc "Vào trước ra trước", giống như việc xếp hàng mua vé, ai xếp trước mua trước. Đối với kiểu dữ liệu Hàng đợi cũng chỉ có hai phép toán là Push và Pull giống như bài toán trên.

Hãy mô tả một kiểu dữ liệu Hàng đợi đơn giản và hai thủ tục Push và Pull.

19. Nguồn gốc số Fibonacci:

Trên một hòn đảo hoang người ta đặt vào một cặp thỏ non (một đực và một cái). Các cặp thỏ non chỉ trưởng thành sau 2 tháng. Khi đã trưởng thành các cặp thỏ sẽ sinh ra một cặp thỏ non sau mỗi tháng. Tính số cặp thỏ trên đảo sau k tháng.

Chứng minh rằng số cặp thỏ cần tìm trên đảo chính bằng số Fibonacci thứ k .

Bài toán trên lần đầu tiên được mô tả trong quyển sách Liber abaci của tác giả Fibonacci người Italia vào thế kỷ thứ 13 và tên gọi số Fibonacci bắt đầu có từ đó.

§4. KHÁI NIỆM HÀM



Câu hỏi, ví dụ, bài tập

1. Viết đoạn chương trình mô tả Function tính giá trị của biểu thức:

$$x^3 - 3x^2 + 1, \text{ ở đây } x \text{ là số thực tùy ý.}$$

2. Tính hàm số a^n , ở đây a là số thực và n là số tự nhiên tùy ý được nhập vào từ bàn phím.

3. Viết Function cho giá trị Boolean kiểm tra tính nguyên tố của một số tự nhiên cho trước.

4. Dùng Function tính UCLN của 2 số tự nhiên x, y theo thuật toán Ôclit.

5. Dùng Function tách từ đầu tiên ra khỏi một chuỗi ký tự cho trước.

6. Viết Function cho giá trị Boolean kiểm tra xem một chuỗi cho trước có phải là một chuỗi đối xứng không.

7. Tính x^y , ở đây $x > 0$ và y là 2 số thực tùy ý được nhập vào từ bàn phím.

8. Dùng Function tính UCLN của k số tự nhiên theo thuật toán Ôclit.

9. Viết lại hàm $\text{Int}(x)$ tính giá trị phần nguyên của số thực x (tức là số nguyên lớn nhất không vượt quá x).

10. Viết hàm tính số nguyên gần số thực x nhất.

11. Viết hàm số tính khoảng cách gần nhất từ số thực x tới các số nguyên.

12. Viết hàm số Euler $\phi(n)$ với n tự nhiên tính số các số tự nhiên nhỏ hơn n và nguyên tố cùng nhau với n .



Bài tập về nhà

13. Viết hàm số tính $n!$

14. Viết hàm số tính $n!!$ (Ký hiệu $n!!$ dùng để chỉ tích $1*3*5*...*n$ nếu n lẻ và $2*4*6*...*n$ nếu n chẵn).

15. Cho $S1, S2$ là hai chuỗi ký tự cùng độ dài. Gọi $d(S1, S2)$ với $S1, S2$ hai chuỗi ký tự cùng độ dài, là tổng số các vị trí khác nhau trong chuỗi.

Ví dụ: $d('abcd', 'acbc')=2$ (các vị trí 2 và 3 khác nhau).

Viết hàm số tính $d(S1, S2)$. Số $d(S1, S2)$ được gọi là Khoảng cách giữa $S1$ và $S2$.

16. Cho trước chuỗi nhị phân S . Viết chương trình liệt kê tất cả các chuỗi nhị phân S' cùng độ dài với S sao cho $d(S, S')=1$.

17. Cho số tự nhiên N . Viết chương trình kiểm tra xem N có thể biểu diễn thành tổng của một số số tự nhiên liên tiếp (> 1 hạng tử) được hay không. Nếu đúng hãy in ra một biểu diễn như vậy.

18. Số tự nhiên N được gọi là "Tích bậc k " nếu nó có thể biểu diễn dưới dạng tích của k số tự nhiên (> 1).

Ví dụ: một số tự nhiên là Tích bậc 2 khi và chỉ khi nó là hợp số.

Cho trước số tự nhiên N , kiểm tra xem N có phải là Tích bậc 3 hay không.

19. Gọi A_k là tập hợp các số tự nhiên là Tích bậc k . Để thấy khẳng định sau:

$$A_1 \supset A_2 \supset A_3 \supset \dots \supset A_k \supset A_{k+1} \supset \dots$$

Số tự nhiên n thuộc A_k nhưng không thuộc A_{k+1} được gọi là có Tích thực sự bậc k . Các số nguyên tố là những số có Tích thực sự bậc 1.

Giả sử số tự nhiên n có dạng khai triển nguyên tố như

sau: $n = p_1^{k_1} p_2^{k_2} \dots p_s^{k_s}$.

Hãy chứng minh rằng khi đó n là số có tích thực sự bậc: $k_1 + k_2 + \dots + k_s$.

20. Viết chương trình nhập số tự nhiên N và tính bậc tích thực sự của nó.

21. Xét hàm số $N()$ xác định trên tập các xâu ký tự chỉ bao gồm các ký tự '0', '1', ..., '9' và được xác định như sau:

$$N("") = 0,$$

$$N('x') = \begin{cases} 1 & \text{nếu } x \text{ chẵn và} \\ -1 & \text{nếu } x \text{ lẻ,} \end{cases}$$

$N(uv) = N(u) + N(v)$, với u, v là các xâu ký tự tương ứng.

a. Tính $N('1233536535')$.

b. Tính $N(u)$ với u là xâu bất kỳ.

22. Cho trước n số tự nhiên $a_1, a_2, \dots, a_n$. Gọi d là UCLN của n số trên. Hãy tìm các số $p_1, p_2, \dots, p_n$ sao cho: $d = p_1 a_1 + p_2 a_2 + \dots + p_n a_n$.

§5. BIẾN CỤC BỘ, BIẾN TỔNG THỂ TRUYỀN THAM BIẾN CHO THỦ TỤC/ HÀM

Câu hỏi, ví dụ, bài tập

1. Các khẳng định sau đúng hay sai:

a. Biến cục bộ là biến được mô tả bên trong thủ tục hoặc hàm.

b. *Biến tổng thể là biến mô tả trong phần khai báo của chương trình chính.*

c. *Biến cục bộ và biến tổng thể trong một chương trình có thể trùng tên.*

2. *Các khẳng định sau đúng hay sai:*

a. *Biến cục bộ không có giá trị ở bên ngoài khi tên của biến đó không được khai báo ở bên ngoài.*

b. *Giá trị của biến cục bộ sẽ được giữ nguyên khi biến đã được khai báo ở bên ngoài.*

c. *Giá trị của biến tổng thể hoàn toàn không phụ thuộc vào việc thực hiện bên trong các hàm và thủ tục.*

3. *Trong Pascal có bao nhiêu cách truyền giá trị cho biến trong Thủ tục và Hàm (phân biệt có và không có từ khóa Var)?*

4. *Đoạn chương trình sau đúng hay sai, vì sao:*

```
Program VD;
    { Phần mô tả của VD }
    (*=====*)
    Procedure Tinh(Var X,Y: real);
    Begin
        { Phần chương trình của thủ tục Tinh }.
    End;
    (*=====*)
    Begin { Chương trình chính }
        Tinh(4,5);
    End.
```

5. Các hàm và thủ tục trong Pascal có thể gọi lồng nhau được hay không?

6. Bạn hãy in và chạy thử 3 chương trình sau đây:

```
a. Program Tham_so;
   Var a, b: Integer;
   (*=====*)
   Procedure VD_thamso(x,y: integer)
   Begin
       x:=x+1; y:= y+1;
       Writeln(x:5,y:5);
   End;
   (*=====*)
   Begin {Chương trình chính}
       a:=1; b:=4;
       VD_thamso(a,b);
       Writeln(a:5, b:5);
   END.
```

Kết quả chạy sẽ là:

2	5
1	4

```
b. Program Tham_so;
   Var a,b: integer;
   (*=====*)
   Procedure VD_thamso(x:integer; Var y: integer)
   Begin
       x:= x+1; y:=y+1;
       Writeln(x:5, y:5);
   End;
   (*=====*)
```

Begin {Chương trình chính}

a:=1; b:=4;

VD_thamso(a,b);

Writeln(a:5, b:5);

End.

Kết quả chạy sẽ là:

2	5
1	5.

c. Program Tham_so;

Var a, b: integer;

(*=====*)

Procedure VD_thamso(Var x, y: integer)

Begin

x:= x+1; y:= y+1;

Writeln (x:5, y:5);

End;

(*=====*)

Begin {chương trình chính}

a:= 1; b:=4;

VD_thamso(a,b);

Writeln(a:5,b:5);

End.

Kết quả chạy sẽ là:

2	5
2	5

Bạn hãy giải thích xem tại sao kết quả lại như vậy (phân biệt tham biến và tham trị).

7. Giả sử ta cần khai báo hai thủ tục (hoặc hàm số) A và B. Giả sử trong phần chương trình của A cần gọi B và

trong phần chương trình của B cần gọi A. Trường hợp trên có thể xử lý trên Pascal được hay không? Vì sao?

8. Trong phần thân của thủ tục hoặc hàm có thể gọi đến chính nó được không?

9. Giải phương trình 2 ẩn sau trong phạm vi các số nguyên: $ax + by = c$, với a, b, c là các số nguyên cho trước. Chương trình chỉ cần chỉ ra một nghiệm nào đó của phương trình trên hoặc đưa ra phủ định về nghiệm.

10. Mọi phân số dương và nhỏ hơn 1 đều có thể biểu diễn (duy nhất!) dưới dạng:

$$a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots + \frac{1}{a_n}}}$$

ở đây $a_1, a_2, \dots, a_n$ là các số tự nhiên.

Viết chương trình nhập hai số tự nhiên p, q từ bàn phím với $p < q$. Sau đó tìm dãy $a_1, a_2, \dots, a_n$ là khai triển của phân số p/q . Kết quả thể hiện trên màn hình.



Bài tập về nhà

11. Viết thủ tục:

Compare (S1, S2: String; Var Kq: String)

thực hiện công việc sau: so sánh hai chuỗi S1 và S2, tìm ra tất cả các ký tự có trong cả hai chuỗi trên. Chuỗi Kq sẽ chứa tất cả các ký tự đó, mỗi ký tự chỉ được nhớ một lần.

12. Cho n xâu ký tự $S_1, S_2, \dots, S_n$. Gọi $d(U, V)$, với U, V là hai xâu ký tự bất kỳ, là tổng số các ký tự không giống nhau trong hai xâu trên, mỗi loại ký tự chỉ được nhớ một lần. Ví dụ $d('aabbcc', 'bccdd') = 2$ vì chỉ có hai ký tự a và d là không giống nhau trong các xâu trên.

a. Viết hàm số tính $d(U, V)$.

b. Tính $\text{Max}(d(S_i, S_j))$ và $\text{Min}(d(S_i, S_j))$.

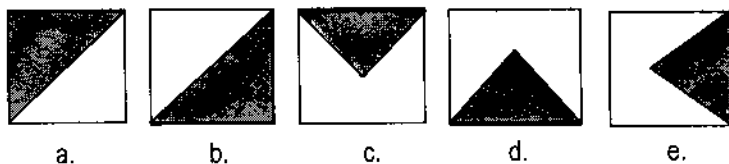
13. Cho một bình hoa có K bông hoa. Hai đôi thủ lần lượt hái.

Luật chơi: Mỗi lần chỉ được hái từ 1 đến M bông hoa ($M < K$ cho trước). Ai hái được bông hoa cuối cùng thì người đó sẽ thắng cuộc.

Hãy thử với $K=24, M=5$.

Lập chương trình để Máy tính chơi với người.

14. Cho bảng số vuông $N \times N$ cho bởi mảng 2 chiều $A[i, j]$. Viết chương trình tính phần tử Max của mảng trên trong các miền sau của bảng số:



15. Hàm Mobius của một số tự nhiên n được định nghĩa như sau:

$M(n) = 1$ nếu $n = 1$ hoặc n biểu diễn thành tích của một số chẵn các số nguyên tố khác nhau, $M(n) = -1$ nếu n biểu diễn thành tích của một số lẻ các số nguyên tố khác nhau và $M(n) = 0$ trong các trường hợp còn lại.

Hãy viết chương trình tính hàm $M(n)$.

16. Ta định nghĩa:

Tổ hợp chập k của n phần tử là số dạng:

$$C(n, k) = \frac{n!}{k!(n-k)!}$$

hay một cách định nghĩa khác $C(n, k)$ là hệ số của khai triển đa thức:

$$(x + y)^n = x^n + C(n, 1)x^{n-1}y + \dots + C(n, k)x^{n-k}y^k + \dots + y^n.$$

Chứng minh các đẳng thức sau đây:

- $C(2n, 2) = 2C(n, 2) + n^2$.
- $C(n, k)C(k, m) = C(n, m)C(n-m, k-m)$.
- $C(m+n, 2) - C(m, 2) - C(n, 2) = mn$.

17. Chứng minh các đẳng thức sau đây:

- $C(p + q, r) = \sum_{j=0}^r C(p, j) C(q, r - j)$
- $C(n + r + 1, r) = \sum_{j=0}^r C(n + j, j)$

18. Gọi một hoán vị $a_1 a_2 \dots a_n$ của $(1, 2, \dots, n)$ là không cân bằng nếu với mọi i thì $a_i < i$. Gọi D_n là số các hoán vị không cân bằng của n phần tử.

Viết chương trình tính D_n và in ra tất cả các hoán vị không cân bằng của n phần tử.

19. Chứng minh rằng dãy số D_n được định nghĩa trong bài trên sẽ thỏa mãn n đẳng thức sau:

$$D_n = (n-1)(D_{n-1} + D_{n-2}) \text{ với } n > 1.$$

20. Chứng minh rằng dãy số D_n được định nghĩa trong bài trên sẽ thỏa mãn n đẳng thức sau:

$$D_n = nD_{n-1} + (-1)^n \quad \text{với mọi } n.$$

§6. LÀM VIỆC VỚI FILE TRONG PASCAL



Câu hỏi, ví dụ, bài tập

1. Một file văn bản trên đĩa có tên là SON.INF chứa đúng một dòng bao gồm hai số tự nhiên n, m được viết cách nhau một dấu cách. Sau đây là một ví dụ về file này:

4 5

Em hãy viết đoạn chương trình nhập hai số này từ file trên và in kết quả ra màn hình.

2. Một file văn bản trên đĩa có tên là SOD.INF chứa đúng hai dòng, mỗi dòng bao gồm đúng một số tự nhiên tương ứng là n, m . Sau đây là một ví dụ về file này:

4

5

Em hãy viết đoạn chương trình nhập hai số này từ file trên và in kết quả ra màn hình.

3. Một file văn bản có tên DATA.TXT lưu trữ số liệu của một mảng n số và có dạng sau:

- Dòng đầu tiên của file ghi số n .

- n dòng tiếp theo của file ghi n số, mỗi số trên một dòng.

Viết chương trình nhập số liệu của dãy trên vào một mảng và in ra dữ liệu của mảng trên sau khi đã sắp lại theo thứ tự tăng dần.

4. Một file văn bản có tên DATA.TXT lưu trữ số liệu của một mảng n số và có dạng sau:

- Dòng đầu tiên của file ghi số n .*
- Dòng tiếp theo của file ghi n số, các số cách nhau bởi tối thiểu một dấu cách.*

Viết chương trình nhập số liệu của dãy trên vào một mảng và in ra dữ liệu của mảng trên.

5. Viết các thủ tục nhập dữ liệu từ file cho các trường hợp của 4 bài trên.

6. Làm bài tương tự bài 3. Điểm khác biệt là ở chỗ file dữ liệu không có dòng đầu tiên ghi tổng số dữ liệu như trong bài 3. Chương trình phải tự kiểm tra và tính số n đó.

7. Làm bài tương tự bài 4. Điểm khác biệt là ở chỗ file dữ liệu không có dòng đầu tiên ghi n , chương trình phải tự kiểm tra và tính số đó.

8. Cho file văn bản ghi số liệu của một bảng số $N \times M$ và có dạng sau:

- Dòng đầu tiên ghi hai số N, M cách nhau bởi dấu cách.*
- N dòng tiếp theo ghi số liệu của N hàng, mỗi dòng bao gồm M số cách nhau bởi dấu cách.*

Sau đây là ví dụ một file như vậy:

4	5			
3	5	-1	12	10
-1	2	3	6	1
1	4	5	10	-11
2	-1	4	5	7

Viết chương trình nhập dữ liệu từ file trên vào một mảng số $N \times M$.

9. Một file văn bản ghi số liệu của học sinh có dạng sau:

4	
Nguyen Van Hung	15 G
Bui Quang Than	14 K
Tran Minh Quang	16 G
Le Van Minh	15 T

Dữ liệu ghi trong file này có ý nghĩa như sau:

- Dòng đầu tiên ghi đúng một số n chỉ số lượng học sinh trong lớp.

- n dòng tiếp theo mỗi dòng ghi dữ liệu của một học sinh.

- Mỗi học sinh được ghi các dữ liệu sau:

Họ và tên: 20 ký tự

Tuổi: 3 ký tự

Xếp loại: 1 ký tự

Hãy viết chương trình nhập dữ liệu học sinh từ file trên và đưa vào các mảng dữ liệu tương ứng, tên file được đọc từ bàn phím.

10. Làm tương tự bài trên, điểm khác biệt là không có dòng đầu tiên ghi số lượng học sinh. Chương trình phải tự nhận biết và tính được số n đó.



Bài tập về nhà

11. Thực hiện tất cả các thao tác với các dạng file văn bản như trong phần Câu hỏi, ví dụ, bài tập nhưng theo chiều ngược lại, tức là thực hiện các thao tác ghi.

12. Có thể được hay không việc thực hiện ghi ra file văn bản trên một dòng các biến với kiểu dữ liệu khác nhau. Ví dụ có thể được hay không việc ghi ra file như sau:

WriteIn(F, Xau1, So1, Xau2, So2);

với Xaux là các biến kiểu string còn Sox là biến kiểu Real.

13. Số liệu được nhập từ một file văn bản INPUT.TXT và có dạng sau:

- Dòng đầu tiên ghi số tự nhiên N ($N < 20$).
- N dòng tiếp theo, mỗi dòng ghi một dãy số, các số cách nhau bởi dấu cách. Số lượng phần tử của các dãy có thể khác nhau.

Viết chương trình nhập số liệu từ file trên, sắp xếp N dãy trên theo thứ tự tăng dần và ghi kết quả ra file OUTPUT.TXT có dạng tương tự như file nhập liệu.

14. Một danh sách lớp được ghi trong một File văn bản có dạng sau:

Nguyen Van Minh	Nam
Nguyen Tung Chau	Nu
Tran Quang Thu	Nam
Bui Van Ngo	Nam

.....

Trong đó họ và tên học sinh được ghi trên độ dài 20 ký tự, 3 ký tự tiếp theo sẽ ghi Nam hoặc Nu chỉ giới tính của học sinh đó. Số dòng của file này sẽ bằng số học sinh trong lớp.

Hãy viết chương trình thực hiện các công việc sau:

- Nhập tên File số liệu từ bàn phím, sau đó nhập dữ liệu từ File vào bộ nhớ máy tính.
- Thực hiện việc sắp xếp lại danh sách học sinh trong lớp sao cho các bạn Nữ lên trước, các bạn Nam sau.
- Nhập tên File kết quả từ bàn phím và ghi danh sách lớp sau khi đã sắp xếp lại ra File đó theo mẫu giống như lúc đọc vào.

15. Đầu bài giống bài trên. Hãy viết chương trình để thực hiện các công việc sau:

- Đếm xem trong lớp có bao nhiêu bạn là Nam, bao nhiêu bạn là Nữ.
- Kiểm tra xem lớp có hai bạn nào cùng họ hay không? Nếu có liệt kê tất cả các bạn có chung họ.

16. Đầu bài giống bài trên. Hãy viết chương trình đọc dữ liệu và in kết quả ra File sau khi đã sắp xếp các bạn trong lớp theo trật tự sau: trước tiên sắp xếp theo giới tính, các bạn Nữ trước, Nam sau, sau đó sắp tiếp trong số các bạn cùng giới theo tên theo thứ tự ABC, sau đó sẽ sắp tiếp theo họ. Trong ví dụ của bài trên file kết quả sẽ có dạng

Nguyen Tung Chau	Nu
Nguyen Van Minh	Nam
Bui Van Ngo	Nam

Tran Quang Thu

Nam

17. Cho trước một xâu nhị phân độ dài bất kỳ được đưa vào từ file văn bản INPUT.TXT. Cần biến đổi xâu nhị phân này về dạng toàn số 0. Các phép biến đổi có thể chỉ là một trong các loại sau:

- Biến đổi xâu con 11 thành 00.
- Biến đổi xâu con 010 thành 000.

Hãy chỉ một cách biến đổi xâu đã cho thành xâu có toàn 0. Kết quả thể hiện trong file OUTPUT.TXT như sau:

Dòng đầu tiên của OUTPUT.TXT chứa xâu ban đầu, sau đó mỗi dòng là một xâu tiếp theo sau một phép biến đổi, xâu cuối cùng là xâu toàn 0.

18. Cho một file văn bản BL.PAS trên đó ghi một chương trình Pascal.

Hãy viết chương trình thực hiện các công việc sau:

- a. Kiểm tra xem trong chương trình trên số các từ Begin có bằng số các từ End hay không.
- b. Kiểm tra tính đúng đắn của các chú thích của chương trình: số các ký tự "{" có bằng số các ký tự "}" không.

19. Một văn bản có tên BL.PAS chứa một chương trình Pascal hoàn chỉnh. Trong chương trình này trên một dòng có thể chứa nhiều lệnh.

Em hãy viết chương trình in ra file mới có tên BLMOI.PAS chương trình Pascal trên sao cho thỏa mãn điều kiện mỗi lệnh phải nằm trên một hàng.

20. Tại một sân ga người ta xét hành lý của hành khách và ghi kết quả vào file có tên KHACH.INF sao cho mỗi hành

khách được ghi trên một dòng, mỗi kiện hàng được ghi rõ số cân trên kiện hàng đó. Ví dụ một file như vậy có dạng sau:

```
4
12  4  5.5  6  2
21
15  5
1   6
```

File trên có ý nghĩa sau: có tất cả 4 hành khách, hành khách 1 có 5 kiện hàng với số cân tương ứng là 12, 4, 5.5, 6 và 2kg, khách thứ hai có 1 kiện nặng 21 kg...

Hãy viết chương trình thực hiện các công việc sau:

a. Tính tổng số cân của mỗi hành khách và in kết quả ra file có tên CAN.KQ, mỗi hành khách là một số trên một hàng. Ví dụ với file đầu vào như trên thì kết quả sẽ là:

```
29.5
21
20
7
```

b. Kiểm tra và in ra những khách hàng không đủ tiêu chuẩn đi tàu. Số liệu được ghi ra file có tên HUY.KQ, mỗi dòng ghi số thứ tự hành khách không đủ điều kiện. Biết rằng hành khách sẽ không đủ điều kiện đi tàu nếu tổng số trọng lượng hàng hóa lớn hơn 20 kg hoặc có quá 10 kiện hàng. Trong ví dụ trên các hành khách số 1 và 2 là không đủ điều kiện do đó file kết quả sẽ có dạng:

```
1
2
```

§7. ÔN TẬP



Câu hỏi, ví dụ, bài tập

1. Viết chương trình con tính diện tích hình tròn, hình vuông, hình thoi trong một chương trình chính. Sau đó hỏi và chọn một số phương án tính diện tích bằng cách chọn trong bảng chọn lệnh sau:

- 1- Tính diện tích hình tròn.
- 2- Tính diện tích hình vuông.
- 3- Tính diện tích hình thoi.

2. Hãy viết lại chương trình giải phương trình bậc 2 bằng cách đưa vào các thủ tục tính giá trị Delta.

3. Hãy viết chương trình cộng 2 số tự nhiên cho trước được nhập vào từ bàn phím với kết quả được viết trong hệ cơ số 2.

4. Tìm và in ra tất cả các số tự nhiên nguyên tố có 3 chữ số là số đối xứng trong cách viết cơ số 2.

5. Lập trình tách từ cuối cùng ra khỏi một xâu ký tự cho trước và kiểm tra xem đó có phải là từ đối xứng không? Và in kết quả ra màn hình.

6. Giải phương trình: $2^n = 3^m + 1$ trong phạm vi các số tự nhiên $1 \leq n, m \leq 10$.

7. Lập trình chuyển một số được viết trong hệ cơ số 9 thành số trong hệ cơ số 3.

8. Các số được viết trong hệ đếm cơ số 16 sử dụng các ký tự sau: 0, 1, ..., 9 và A, B, C, D, E, F chỉ các số 10, 11, 12, 13, 14, 15.

Hãy viết chương trình nhập số tự nhiên N từ bàn phím và chuyển số đã cho sang hệ đếm 16 (còn được gọi là hệ hexa).

9. Lập chương trình thực hiện theo thực đơn sau:

- 1- Nhập số tự nhiên N.
- 2- Nhập số M trong hệ đếm 16.
- 3- Chuyển N sang hệ 16.
- 4- Chuyển M sang hệ 10.
- 5- Ghi kết quả ra file.

Chú ý: Tập kết quả sẽ có tên KQ12.TXT và bao gồm 4 dòng. Dòng thứ nhất ghi số N dạng thập phân, dòng thứ 2 ghi số M dưới dạng Hexa, dòng thứ 3 ghi N dạng hexa, dòng thứ 4 ghi M dạng thập phân.



Bài tập về nhà

10. Cho trước các hệ số nguyên $a_0, a_1, \dots, a_n$. Giải phương trình sau trong phạm vi số nguyên:

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 = 0.$$

11. Thuật toán cộng số nguyên:

Xét các biểu diễn nhị phân của các số tự nhiên:

$$a = a_{n-1}a_{n-2} \dots a_1a_0,$$

hay có thể viết $a = (a_{n-1}a_{n-2} \dots a_1a_0)_2$.

Ta có thuật toán sau mô tả phép cộng các số nguyên theo cách biểu diễn nhị phân:

Giả sử $a = a_{n-1}a_{n-2}...a_1a_0$, $b = b_{n-1}b_{n-2}...b_1b_0$ và kết quả của phép cộng:

$$a + b = c = c_n c_{n-1} c_{n-2} ... c_1 c_0.$$

Thuật toán cộng số nhị phân như sau:

```

e:=0
for i:=0 to n-1 do
  Begin
    f:=int((ai + bi + e)/2)
    ci:= ai + bi - 2f
    e:=f
  End
cn:= e
    
```

Hãy chứng minh tính đúng đắn và mô tả thuật toán trên bằng chương trình Pascal.

12. Thuật toán nhân số nguyên: Để mô tả thuật toán nhân các số nguyên bằng biểu diễn nhị phân ta cần định nghĩa hai phép toán sau:

Giả sử $a = a_{n-1}a_{n-2}...a_1a_0$ và $b = b_{n-1}b_{n-2}...b_1b_0$ khi đó định nghĩa phép toán **left** (chuyển sang trái) và **right** (chuyển sang phải) một số bit như sau:

a **left** k = $a_{n-k-1}...a_1a_000...0$ (k số 0 bên phải)

b **right** k = $00...0a_{n-1}a_{n-2}...a_{n-k}$ (k số 0 bên trái).

Để mô tả thuật toán ta sử dụng các biến trung gian là $c_0, c_1, ..., c_{n-1}$ (các số nguyên).

Kết quả sẽ được gán cho biến p .

for $i:=0$ to $n-1$ do

Begin

if $b_i = 1$ then $c_i := a$ left i

Else $c_i := 0$

End;

$p:=0$;

for $i:=0$ to $n-1$ do $p:=p+c_i$

Viết chương trình mô tả thuật toán trên.

13. Thuật toán nhân nhanh:

Trong bài tập này sẽ mô tả một thuật toán nhân rất nhanh các số nguyên.

Giả sử ta có: $a = a_{2n-1}a_{2n-2}...a_1a_0$ và $b = b_{2n-1}b_{2n-2}...b_1b_0$ là các biểu diễn nhị phân của a và b . Chú ý rằng ta có thể coi chúng có cùng độ dài $2n$ bằng cách bổ sung các bit 0 về bên trái của các số đó.

Đặt $a = 2^n A_1 + A_0$ và $b = 2^n B_1 + B_0$

Ở đây $A_1 = a_{2n-1}a_{2n-2}...a_{n+1}a_n$, $A_0 = a_{n-1}a_{n-2}...a_1a_0$,

$B_1 = b_{2n-1}b_{2n-2}...b_{n+1}b_n$, $B_0 = b_{n-1}b_{n-2}...b_1b_0$.

Khi đó thuật toán nhân nhanh được thể hiện trong công thức sau:

$$ab = (2^{2n} + 2^n)A_1B_1 + 2^n(A_1 - A_0)(B_0 - B_1) + (2^n + 1)A_0B_0.$$

a. Chứng minh tính đúng đắn của công thức trên.

b. Viết chương trình mô tả thuật toán trên.

14. Chứng minh rằng dãy số Fibonacci $F_1, F_2, ..., F_n, ...$ thỏa mãn đẳng thức sau:

$$F_k F_n + F_{k+1} F_{n+1} = F_{n+k+1} \text{ với mọi số tự nhiên } k, n.$$

15. Chứng minh các đẳng thức sau với số Fibonacci:

$$a. F_1^2 + F_2^2 + \dots + F_n^2 = F_n F_{n+1}.$$

$$b. F_1 + F_3 + \dots + F_{2n-1} = F_{2n}.$$

16. Dãy số Lucas được định nghĩa như sau: $L_0=2$, $L_1=1$ và $L_n = L_{n-1} + L_{n-2}$ với $n > 1$.

a. Chứng minh liên hệ sau giữa các số Lucas và Fibonacci: $F_n + F_{n+2} = L_{n+1}$.

b. Chứng minh hằng đẳng thức:

$$L_0^2 + L_1^2 + \dots + L_n^2 = L_n L_{n+1} + 2$$

17. Chứng minh các đẳng thức sau:

$$a. F_{n+1} F_{n-1} - F_n^2 = (-1)^n.$$

$$b. F_0 F_1 + F_1 F_2 + \dots + F_{2n-1} F_{2n} = F_{2n}^2.$$

$$c. F_0 - F_1 + F_2 - F_3 + \dots - F_{2n-1} + F_{2n} = F_{2n-1} - 1.$$

18. Dãy số a_n được định nghĩa qui nạp như sau:

$$a_1 = 1$$

$$a_2 = 2.$$

Giả sử tất cả các số tự nhiên nhỏ hơn n đều đã được xác định là thuộc dãy trên hay không.

Khi đó số n sẽ được xác định thuộc dãy trên nếu nó có thể biểu diễn duy nhất dưới dạng tổng của hai số khác nhau của dãy trên.

Ta có ví dụ $a_3 = 3$, $a_4 = 4$, $a_5 = 6$, ...

a. Em hãy tính tay 10 số hạng đầu tiên của dãy trên.

b. Lập chương trình tính 100 số hạng đầu tiên của dãy y.

19. Viết chương trình thực hiện các công việc sau:

a. Nhập số tự nhiên n ($n < 3000$) và tính xem năm n có bao nhiêu chủ nhật.

b. Nhập hai ngày $D1(dd/mm/yyyy)$ và $D2(dd/mm/yyyy)$ và tính xem có bao nhiêu ngày giữa $D1$ và $D2$ (giả sử $D1 < D2$).

c. Tính xem trong thế kỷ 19 có bao nhiêu ngày thứ 7 và chủ nhật.

d. Hãy kiểm tra điều các sách lịch sử vẫn nói rằng ngày 1 tháng 1 năm 1 (ngày đầu tiên của Công nguyên) là thứ hai.

§8. TỔNG ÔN TẬP CHƯƠNG TRÌNH



Câu hỏi, ví dụ, bài tập

1. Viết chương trình nhập số tự nhiên N và in ra chữ số thứ N của dãy vô hạn các số nguyên không âm chẵn: 0246810121416182022...

2. Viết chương trình nhập số tự nhiên N và in ra chữ số thứ N của dãy vô hạn các số tự nhiên lẻ: 1357911131517192123...

3. Viết chương trình nhập số tự nhiên N và in ra chữ số thứ N của dãy vô hạn các số chính phương: 1491625...

4. Viết chương trình nhập số tự nhiên N và in ra chữ số thứ N của dãy vô hạn các số Fibonacci.

5. Các dãy số a_n, b_n được cho bởi các đẳng thức liên hệ sau:

$$a_0 = 1, b_0 = 2$$

$$a_n = a_{n-1} + b_{n-1}$$

$$b_n = a_{n-1} - b_{n-1}$$

Viết chương trình nhập n từ bàn phím và in ra kết quả là $n + 1$ số hạng đầu tiên của hai dãy trên. Kết quả thể hiện trong file văn bản KQ.TXT có dạng: dòng đầu tiên ghi số n , $n + 1$ dòng tiếp theo ghi lần lượt hai số a, b tương ứng, hai số này cách nhau bởi dấu cách.

6. Viết chương trình sinh ngẫu nhiên một xâu ký tự độ dài N .

7. Lập chương trình tính xem trong một ngày từ 0 giờ sáng đến 24 giờ đêm kim giờ và kim phút gặp nhau mấy lần.

8. Xét thuật toán sau dùng để tính các hệ số p, q từ phương trình $ap + bq = d$, với $d = \text{UCLN}(a, b)$.

Đặt $c_0 = a, c_1 = b$ và xét dãy các phép biến đổi Euclid như sau để sinh ra dãy $c_1, c_2, \dots, c_n$:

$$c_0 = r_1 c_1 + c_2$$

$$c_1 = r_2 c_2 + c_3$$

.....

$$c_{n-2} = r_{n-1} c_{n-1} + c_n$$

$$c_{n-1} = r_n c_n$$

Ta sẽ có $d = c_n$

Thuật toán dùng để tính p, q như sau:

Ta lần lượt biểu diễn $c_0, c_1, \dots, c_n$ qua a và b .

Với c_0 ta có: $c_0 = 1.a + 0.b$

Với c_1 ta có: $c_1 = 0.a + 1.b$

...

Giả sử ta đã có: c_i và c_{i+1} biểu diễn được qua a và b .
 Khi đó do $c_{i+2} = c_i - r_{i+1}c_{i+1}$ nên dễ dàng suy ra c_{i+2} sẽ biểu diễn được qua a và b .

Hãy biểu diễn thuật toán trên thành chương trình Pascal hoàn chỉnh.

9. Cho trước hai số tự nhiên nguyên tố cùng nhau A và p . Tìm thuật toán chỉ ra số tự nhiên $B < p$ sao cho $AB \bmod p = 1$. (Gợi ý: sử dụng kết quả của bài trên).

10. Định lý Trung quốc về lý thuyết số phát biểu như sau:

Cho trước các số tự nhiên nguyên tố cùng nhau $P_1, P_2, \dots, P_m$. Giả sử $a_1 < P_1, a_2 < P_2, \dots, a_m < P_m$ các số nguyên không âm bất kỳ.

Khi đó tồn tại duy nhất một số tự nhiên $N < P_1 P_2 \dots P_m$ sao cho $N \bmod P_i = a_i$ với $i = 1, 2, \dots, m$.

Ta có thuật toán sau để thực hiện định lý trên:

Gọi $Q_1 = P_2 P_3 P_4 \dots P_m$

$Q_2 = P_1 P_3 P_4 \dots P_m$

...

$Q_i = P_1 P_2 \dots P_m$ (không có P_i)

...

Với mỗi i ta chọn số tự nhiên R_i sao cho:

$$R_i < P_i \text{ và } R_i Q_i \bmod P_i = 1$$

Kết quả của bài toán trên cho ta cách tìm các R_i như vậy. Khi đó:

$$N = Q_1 R_1 a_1 + Q_2 R_2 a_2 + \dots + Q_m R_m a_m \bmod P_1 P_2 \dots P_m$$

chính là số cần tìm.

a. Em hãy chứng minh thuật toán trên dẫn đến kết quả đúng.

b. Viết chương trình thực hiện bài toán trên như sau:

Các số tự nhiên $P_1, P_2, \dots, P_m$ được đưa vào từ File văn bản có tên IN.INP có dạng: dòng đầu tiên ghi số m , m dòng tiếp theo mỗi dòng ghi một số P_i .

Chương trình sẽ yêu cầu nhập $a_1, a_2, \dots, a_m$ từ bàn phím và sẽ in số N ra màn hình.



Bài tập về nhà

11. Viết chương trình thực hiện trò chơi sau:

Có N số 1, 2, 3, ..., N được lần lượt xếp trên vòng tròn theo chiều kim đồng hồ. Các số tự nhiên N, k được đọc vào từ bàn phím ngay từ đầu trò chơi. Mỗi lần thực hiện trò chơi, máy sẽ làm các việc sau:

- Yêu cầu người chơi nhập một số tự nhiên A ($1 \leq A \leq N$). Đây chính là số mà người chơi muốn máy xóa đi chậm nhất.

- Bắt đầu từ số 1 máy bắt đầu đi dạo theo chiều kim đồng hồ, cứ đến số thứ k máy sẽ xóa số đó đi khỏi vòng tròn và thông báo ngay với người chơi.

- Chương trình dừng khi máy xóa đến số A , máy sẽ thông báo số điểm mà bạn đạt được: xóa mỗi số trước số A sẽ được một điểm. Như vậy bạn chơi càng giỏi thì điểm càng cao.

12. Trò chơi đoán chữ (1):

Máy tính sẽ sinh ngẫu nhiên một xâu ký tự độ dài n ($n < 10$) và thông báo số n cho bạn. Yêu cầu xâu chỉ bao gồm các chữ cái tiếng Anh in hoa.

Bạn có nhiệm vụ đoán xâu ký tự đó.

Máy sẽ lần lượt yêu cầu bạn nhập xâu ký tự phải đoán, máy sẽ thông báo hai số:

- Số các chữ cái đúng nhưng sai vị trí.
- Số các chữ cái đúng vị trí.

Trong khi bạn tìm, kết quả các lần thử trước đều hiện trên màn hình. Khi bạn đã đoán đúng chương trình sẽ dừng lại thông báo số lần bạn thử.

13. Trò chơi đoán chữ (2):

Điều kiện của trò chơi tương tự bài trên, điểm khác là ở các giá trị số do máy đưa ra:

- Số các chữ cái đúng.
- Số các chữ cái không đúng.

14. Trò chơi cân bi:

Có 12 hòn bi giống hệt nhau về hình dáng và khối lượng. Tuy nhiên có một hòn bi trong số đó có khối lượng khác thường (nhẹ hoặc nặng hơn). Có một cái cân đôi. Chỉ sau 3 lần cân sẽ tìm được hòn bi đó và nói chính xác là nhẹ hơn hay nặng hơn.

Bạn là người ra câu đố, máy phải tìm cách cân. Các hòn bi sẽ được đánh số từ 1 đến 12.

Cách chơi: Bạn sẽ nghĩ trong đầu hòn bi không bình thường đó và phải trả lời các câu hỏi của máy. Máy sẽ tiến hành các lượt cân thử, căn cứ vào giả thiết của mình bạn phải trả lời kết quả của mỗi lần cân.

Sau 3 câu trả lời máy sẽ tìm ngay hòn bi không bình thường đó và bạn chỉ việc so sánh với giả thiết của mình.

15. Tìm tất cả các nghiệm nguyên không âm của phương trình:

$$x_1 + x_2 + \dots + x_m = n$$

thỏa mã $n, x_1 < x_2 < \dots < x_m$. Các số tự nhiên n, m được nhập từ file IN.TXT. Kết quả của chương trình được ghi ra file OUT.TXT thành nhiều dòng, mỗi dòng là một nghiệm của phương trình.

16. Tìm tất cả các nghiệm nguyên không âm của phương trình:

$$x_1 + x_2 + \dots + x_m + \dots = N$$

thỏa mã $n, x_1 < x_2 < \dots < x_m < \dots$. Số tự nhiên N được nhập từ bàn phím. Kết quả của chương trình được ghi ra file OUT.TXT thành nhiều dòng, mỗi dòng là một nghiệm của phương trình.

17. Các số nguyên "lớn" được mô tả bởi mảng ký tự sau:

Array[0..MaxInt] of byte.

Hãy viết chương trình:

- Nhập một số nguyên lớn từ bàn phím.
- Thực hiện phép nhân số lớn đó với một số nguyên được nhập từ bàn phím.

18. Sử dụng kết quả của bài trên tính $N!$ với N là số tự nhiên nhập từ bàn phím.

19. Mô tả phép chia nguyên (div) của các số lớn với số tự nhiên.

20. Cho trước một số nguyên lớn. Hãy chuyển đổi số này sang cách viết nhị phân.

Phần 2. Hướng dẫn

§1. ÔN TẬP

DỮ LIỆU VÀ THUẬT TOÁN TRONG PASCAL

1. Hai kiểu khai báo sau:

Var Ch: Char;

Var St: String[1];

là không tương đương.

a. Có thể thực hiện lệnh gán là: St:=ch;

b. Không thể thực hiện lệnh gán Ch:=st;

2. Trong một chương trình Pascal nếu ta khai báo kiểu biến số nhưng không đặt giá trị ban đầu cho chúng thì các biến này sẽ được ngầm định như sau:

- Nếu là biến kiểu nguyên thì nó sẽ nhận giá trị bằng 0.

- Nếu là biến thực thì nó sẽ nhận giá trị bằng 0.0.

- Nếu là biến kiểu chuỗi ký tự thì nó nhận giá trị là một chuỗi rỗng.

- Nếu là biến kiểu ký tự thì nó nhận giá trị là ký tự có mã bằng 0.

3. Program P4103;

Uses Crt;

Var a: array[0..9] of byte;

i,j: byte;

Begin

Clrscr;

```

Writeln(' Tam giac Pascal :');
for i:=0 to 9 do
  Begin
    For j:=i downto 0 do
      If (j=i)or(j=0) then a[j]:=1
      Else a[j]:=a[j]+a[j-1];
      Gotoxy((70-5*i) div 2,i+8);
      For j:=0 to i do Write(a[j]:5);
    End;
  Readln;
End.

```

4. Program P4104;
 Uses Crt;
 Var a: array [0..9] of byte;
 i, j: byte;
 Begin
 Clrscr;
 Writeln('Tam giác Pascal :');
 Writeln; Writeln;
 For i:=0 to 9 do
 Begin
 For j:=i downto 0 do
 If (j=i)or(j=0) then a[j]:=1
 Else a[j]:=a[j]+a[j-1];
 For j:=0 to i do Write(a[j]:5);
 Writeln;
 End;

Readln;

End.

5. a. X=0.1;
b. X=1;
c. X=0.1;

6. Chương trình sẽ liệt kê tất cả các số nhị phân N bit. (N được nhập từ bàn phím).

Thứ tự các bit như sau:



7. Program P4107;
Uses Crt;
Const dau=['+', '-'];
 so=['0'..'9'];
Var st: string;
 i: byte;
 (*=====*)
Function kiem_tra(SS: string): boolean;
 { Kiểm tra xem xâu SS có biểu diễn đúng 1 biểu thức số học không }
 Var kt: boolean;
 d, len: byte;
 Begin
 kt:=true;
 len:=length(SS);
 If kt then
 For i:=2 to len-1 do
 If not (SS[i] in so+dau+['.']) then kt:=false;

```

    If (not(SS[1] in so+dau)) or (not(SS[len] in so)) then
        kt:=false;
    i:=1;
    d:=0;
    While (i<len)and kt.do
        If (SS[i] in dau+'.'))and(ss[i+1] in dau+'.')) then
            kt:=false
        Else
            Begin
                If SS[i]='.' then inc(d);
                If d>=2 then kt:=false;
                If SS[i] in dau then d:=0;
                inc(i);
            End;
        kiem_tra:=kt;
    End;
    (*=====)
    Function tinhgt(ss:string):real;
    Var cd: integer;
        s: string;
        T, num: real;
    Begin
        i:=0;
        s:=""; { Xấu rỗng }
        T:=0;
        While i<=length(ss) do
            Begin
                inc(i);

```

```

    If not (ss[i] in dau) and (i<=length(ss)) then
        s:=concat(s,ss[i])
    Else
        Begin
            Val(s,num,cd);
            T:=T+num;
            s:=ss[i];
        End;
    End;{ While... }
    Tinhgt:=T;
End;
(*=====*)
BEGIN { Main Program }
    Clrscr;
    Writeln('Nhập xâu ST dạng X1 ± X2 ± X3 ±...± Xn,');
    Writeln('trong đó X1,X2,...,Xn là các số nguyên '
        + 'hoặc số thực:');
    Write('ST='); Readln(st);
    If kiem_tra(st) then
        Begin
            Writeln('Giá trị của biểu thức ST bằng: ');
            Writeln(tinhgt(st):10:2);
        End
    Else Writeln('Xâu "',st,'" không biểu diễn đúng'
        + 'biểu thức của bài toán. ');
    Write('Nhấn Enter để thoát...');
    Readln;
End.

```

8. Đúng.

a. Thay While bằng Repeat

Ví dụ: Đoạn chương trình tính: $S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} + \dots$

cho đến khi $1/n \leq \epsilon$ (ϵ rất nhỏ, được nhập từ bàn phím).

Dùng While:

```
...
S:=0; i:=1;
Write ('Epsilon='); Readln(E);
While (1/i > E) do
  Begin
    S:=S+1/i;
    i:=i+1;
  End;
...
```

Thay bằng Repeat :

```
...
S:=0; i:=0;
Write('Epsilon='); Readln(E);
Repeat
  inc(i);
  If 1/i > E then S:=S+1/i;
Until 1/i <= E;
...
```

b. Thay Repeat bằng While:

Ví dụ 2: Đoạn chương trình tìm ước số chung lớn nhất của hai số a và b dùng Repeat:

...


```
Repeat
  If b>0 then
    Begin
      r:=a mod b;
      a:=b; b:=r;
    End;
Until b=0;
USCLN:=a;
```

...

Thay bằng While :

...

```
While b>0 do
  Begin
    r:=a mod b;
    a:=b; b:=r;
  End;
USCLN:=a;
```

...

9. Một lệnh For có thể thay thế bằng lệnh While nhưng điều ngược lại thì có thể không đúng.

10. Program P4110;

```
Var i, a: integer;
    Stop: boolean;
Begin
  i:=1;
  Stop:=False; {chưa dừng}
  While (i<=100) and (not Stop) do
```

```

Begin
    Write ('Nhập a:'); Readln(a);
    If a<= 0 then
        {do nothing}
    Else If a mod 5 = 0 then
        Begin
            Writeln(a);
            Stop:=True;
        End;
        inc(i);
    End;
    If not stop then Writeln('Không tìm thấy');
End.

```

11. Program P4111;
- Uses Crt;
- Var a: array[0..33] of longint;
- i, j, n: 0..33; { n>33 thì he so > maxlongint }
- Begin
- Clrscr;
- Writeln('Tam giác Pascal:');
- Write('Nhập số n:'); Readln(n);
- For i:=0 to n do
- Begin
- For j:=i downto 0 do
- If (j=i) or (j=0) then a[j]:=1
- Else a[j]:=a[j]+a[j-1];
- End;

```
    For j:=0 to n do Write(a[j]:10);  
    Readln;  
End.
```

12. Program P4112;

```
Uses Crt;  
Var ih, jh: array[1..8] of byte;  
    i, j, k: byte; ihm, jhm: integer;  
    dl, ok, t: boolean;  
Begin  
    Clrscr;  
    Writeln('Nhập vị trí của k quân hậu trên bàn cờ'  
            +'('1<=k,ik,jk<=8):'); Readln(k);  
    For i:=1 to k do  
        Begin  
            Write('(i,i,',',',',i,')='); Readln(ih[i],jh[i]);  
        End;  
    dl:=true;  
    For i:=1 to k do  
        For j:=1 to k do  
            If (i<>j)and((ih[i]=ih[j])or(jh[i]=jh[j])  
                or(abs(ih[i]-ih[j])=abs(jh[i]-jh[j]))) then dl:=false;  
    If not(dl) then  
        Writeln('Các con hậu trên không ở trong trạng thái'  
                +'độc lập.')    Else  
        Begin  
            Writeln(' Các con hậu trên ở trong trạng thái '  
                    +'độc lập.');
```

```

ok:=false;
ihm:=1;
Repeat
    jhm:=1;
    Repeat
        t:=false;
        For i:=1 to k do
            If (ihm =ih[i])and(jhm =jh[i]) then t:=true;
            If not(T) then
                Begin
                    ok:=true;
                    For i:=1 to k do
                        If((ih[i]=ihm)or(jh[i]=jhm)
                        or (abs(ih[i]-ihm)=abs(jh[i]-jhm)))
                            then ok:=false;
                    End;
                    inc(jhm);
                Until (jhm>8)or(ok);
                inc(ihm);
            Until (ihm>8)or(ok);
        If ok then
            Writeln(' Có thể bổ sung thêm một quân hậu nữa'
                +' vào vị trí',ihm-1:3, jhm-1:3)
        Else
            Writeln(' Không thể bổ sung thêm một quân'
                +'hậu nào nữa. ');
    End;
Readln;
End.

```

```

13. Program P4113;
Uses Crt;
Var it, jt: array[1..8] of byte;      {Vị trí các con tốt}
    k, kc: byte; {kc: số lượng các con tốt bị khống chế}
    tot, stepi, stepj: integer;
    ih0, jh0, ih, jh: integer; {Vị trí con hậu và vị trí bị nó
                                khống chế}
    t, ok, co: boolean;
Begin
  Clrscr;
  Writeln('Nhập vị trí của k quân tốt trên bàn cờ'
    + '(1<= k, ik, jk <=8):');
  Write(' k = '); Readln(k);
  For tot:=1 to k do
    Begin
      Write(' (i', tot, ', ', j', tot, ') = '); Readln(it[tot], jt[tot]);
    End;
  co:=false;
  For ih0:=1 to 8 do
    For jh0:=1 to 8 do {Kiểm tra cả 64 ô bàn cờ}
      Begin
        kc:=0;
        t:=false; {Kiểm tra vị trí đó có phải là vị trí
                  con tốt hay không}
        For tot:=1 to k do
          If (it[tot]=ih0) and (jt[tot]=jh0) then t:=true;
        If not(t) then
          Begin

```

```

For stepi:=-1 to 1 do
For stepj:=-1 to 1 do
    {Kiểm tra cả tám hướng của con hậu}
Begin
    ih:=ih0;
    jh:=jh0;
    ok:=false;
    ih:=ih+stepi;
    jh:=jh+stepj;
    While (ih<=8)and(ih>=1)and(jh<=8)
    and(jh>=1) and ((stepi<>0)
    or(stepj<>0))and(not ok) do
        Begin
            For tot:=1 to k do
                If (it[tot]=ih)and(jt[tot]=jh) then
                    Begin
                        ok:=true; kc:=kc+1;
                    End;
                ih:=ih+stepi;
                jh:=jh+stepj;
            End;
        End; {For step}
    If (kc=k) then
        Begin
            co:=true; WriteLn(ih0,jh0:4);
        End;
    End; { If not(t) }
End; {For ih0, jh0}

```

```

    If not(co) then Writeln(' Không có vị trí nào. ');
    Readln;
End.

```

14. Program P4114;

```

Uses Crt;

Var q1, qlmin: array [1..32] of byte;  {Mảng ghi kết quả}
    t: array['I'..'J', 1..64] of byte;  {Vị trí các con tốt}
    h: array[1..64] of set of byte;     {Khả năng của các
                                         con hậu}
    sx: array[1..64] of byte;           {Sắp xếp các con hậu
                                         theo khả năng}

    s, st: set of byte;
    i, j, ih, jh, jmax, k, kc, kcmx: integer;
    l, tot, stepi, stepj, kq, kqmin: integer;
    trung, ok, stop: boolean;

Begin
    Clrscr;
    Writeln ('Nhập vị trí thứ k của quân tốt trên bàn cờ',
              + '(1<=k<=63, 1<=ik, jk<=8):');
    Write('k = '); Readln(k);
    For tot:=1 to k do
        Begin
            Write('(i', tot, ', ', j, ', tot, ') = ');
            Readln(t[i, tot], t[j, tot]);
            st:=st+[tot];
        End;
    {Kiểm tra xem những con hậu nào không chế được
    những con tốt nào}

```

```

For i:=1 to 8 do
  For j:=1 to 8 do {kiểm tra cả 64 ô của bàn cờ}
    Begin
      trung:=false;
      {Kiểm tra vị trí đó có phải là vị trí của con
      tốt hay không}
      For tot:=1 to k do
        If (t[i,tot]=i)and(t[j,tot]=j) then
          trung:=true;
      If not(trung) then
        Begin
          For stepi:=-1 to 1 do
            For stepj:=-1 to 1 do
              {kiểm tra cả tám hướng của con hậu}
              Begin
                ih:=i;
                jh:=j;
                ok:=false;
                ih:=ih+stepi;
                jh:=jh+stepj;
                While(ih<=8)and(ih>=1)and(jh<=8)
                  and(jh>=1) and((stepi<>0)
                  or(stepj<>0))and(not ok) do
                    Begin
                      For tot:=1 to k do
                        If (t[i,tot]=ih)and(t[j,tot]=jh) then
                          Begin
                            ok:=true;

```



```

    jmax:=i;
    For j:=i+1 to 64 do
        Begin
            kc:=0;
            For tot:=1 to k do
                If tot in h[j] then kc:=kc+1;
            If (kc>kamax) then
                Begin
                    kmax:=kc;
                    jmax:=j;
                End;
            End;
        s:=h[i];
        h[i]:=h[jmax];
        h[jmax]:=s;
        l:=sx[i];
        sx[i]:=sx[jmax];
        sx[jmax]:=l;
    End;
    {Tìm một lời giải đầu tiên}
    kqmin:=64;
    i:=1;
    kq:=1;
    ql[kq]:=i;
    stop:=false;
    Repeat
        s:=[];
        For j:=1 to kq do s:=s+h[ql[j]];

```

```

While (s<>st)and(i<64-k) do
  Begin
    inc(i);
    If (h[i]-s<>[]) then
      Begin
        kq:=kq+1;
        q[kq]:=i; s:=s+h[i];
      End;
    End;
  If (s=st) then
    If (kqmin>kq) then {So sánh với lời giải đã có}
      Begin
        kqmin:=kq;
        For j:=1 to kqmin do qlmin[j]:=q[j];
      End;
    If ( kq=0) then stop:=true;
    i:=q[kq];
    dec(kq);
  Until stop;
  Writeln('Số tối thiểu các con hậu phải xếp là: ',
    kqmin, ' con. ');
  Write('Vị trí của các con hậu như sau:');
  For j:=1 to kqmin do
    Writeln('(i,j,j,.)=',(sx[qlmin[j]]-1) div 8+1,',',(sx[qlmin[j]]-1)
      mod 8+1,');
  End;    {Tìm lời giải}
  Readln;
End.

```

```
15. Program P4115;
    Uses Crt;
    Const Max=100;
    Type Bit=0..1;
           Spt=1..Max;
    Var  a: array[Spt] of Bit;
           j, k, i, n: Spt;
    (*=====*)
    Procedure nhap(Var nn: Spt);
    Begin
        Repeat
            Write('N=');Readln(nn);
            If (nn<=0)or(nn>100) then Write(#7,'Nhập lại. ');
        Until (0<nn)and(nn<=100);
        For i:=1 to nn do a[i]:=random(2);
    End;
    (*=====*)
    Procedure in_day(nn: Spt);
    Begin
        For i:=1 to nn do Write(a[i], ' '); Writeln;
    End;
    (*=====*)
    Procedure doi(Var x, y: Bit);
    Begin
        x:=(x+1) mod 2;
        y:=(y+1) mod 2;
    End;
    (*=====*)
```

```

Procedure bien_doi(nn: Spt);
Begin
  For i:=1 to nn do
    Begin
      If (i=1)and(a[nn]<>a[2]) then doi(a[nn],a[2]);
      If (i=nn)and(a[i-1]<>a[1]) then doi(a[i-1],a[1]);
      If (1<i)and(i<nn)and(a[i-1]<>a[i+1]) then
        doi(a[i-1],a[i+1]);
    End;
  End;
  End;
  (*=====*)
  Begin { Main Program }
    Clrscr;
    Writeln('nhập số phần tử của dãy là N với 0<N<=100');
    nhap(n);
    Write('Số phút k='); Readln(k);
    in_day(n);
    Repeat
      bien_doi(n);
    Until(j>k);
    Writeln('Sau khi biến đổi ta được dãy ');
    in_day(n);
    Write('Nhấn Enter để thoát...');
    Readln;
  End.

```

16. a. 1 0 1 0 1 0 1 0 1 0 1 0

Dãy này không bị biến đổi nữa nên quá trình biến đổi bị dừng lại.

b. 100100100100
111111111111

Dãy này không bị biến đổi nữa nên quá trình biến đổi bị dừng lại.

17. a. 1010101000101100
1010101101100011
0010100000010101
1110110000110101
1100001001000100
0010011111101111
1111101111000110
0111000110101000
1010101000101100

Trùng với dãy đầu tiên.

Vậy dãy ban đầu là tuần hoàn.

b. 0100010101111001
0110110100110111
0000000111000010
0000001010100111
1000011010111010
1100100010010010
0011110111111110
0101100011111101

Trùng với dãy đầu tiên.

Vậy dãy ban đầu là tuần hoàn.

18. a. 24 bao gạo

1	5	1
5		5
1	5	1

b. 16 bao gạo

4	1	2
1		1
2	1	4

19. Ma trận kỳ ảo bậc 4

1	14	15	4
12	7	6	9
8	11	10	5
13	2	3	16

20. Ma trận kỳ ảo bậc 5

1	6	15	10	21
14	9	20	5	16
19	2	7	22	11
8	13	24	17	4
25	18	3	12	23

§2. KHÁI NIỆM THỦ TỤC

1. Được.

Ví dụ:

```
Procedure Print (St: String; x, y: byte; mauchu, maunen:
                byte);
```

```
(*=====*)
```

```
Procedure Datmau;
```

```
Begin
```

```
    Textcolor(mauchu);
```

```
    Textbackground(maunen);
```

```

End;
(*=====*)
Begin      {thân của thủ tục Print}
    Gotoxy(x, y);
    Datmau;
    Write(st);
End;

```

2. a. Được sử dụng
- b. Không

```

3. Program P4203;
    Uses Crt;
    Var a, b: integer;
    (*=====*)
    Procedure Input (Var X, Y: integer);
    Begin
        Readln(X);
        Readln(Y);
    End;
    (*=====*)
    Begin
        Clrscr;
        Write('Nhập A, B='); Input(A, B);
        Write('Thương A/B=', A/B);
        Readln;
    End.

```

4. Program P4204;
- Uses Crt;


```

Type   mang = array[1 .. 100] of Real;
Var    a: mang;
(*=====*)
Procedure input (ten: Char; Var b: mang;
                so phantu: byte);

Var i: byte;
Begin
    For i:=1 to sophantu do
        Begin
            Write(Ten, '[', i, ']='); Readln(b[i]); End;
        End;
    End;
(*=====*)
Begin    {Main program}
    Clrscr;
    Input('A', a, 10);
    For i:=1 to 10 do Write(a[i]:6:2);
    Readln;
End.

```

5. Program P4205

```

Uses Crt;
Const M=10; N=20;
Type   A=array[1..N, 1..M] of Real;
        AT=array[1..M, 1..N] of Real;
        {ma trận chuyển vị của ma trận A}
Var    A1: A; AT1:AT;
        m1, n1: byte;
        i, j: byte;

```

```

(*=====*)
Procedure ChuyenVi(X: A, m1, n1: byte; Var Y: AT);
{chuyển vị ma trận X có m1 hàng, n1 cột thành ma trận Y
có n1 cột, m1 hàng}
Begin {bắt đầu thủ tục chuyển vị}
    For i:=1 to n1 do
        For j:=1 to m1 do Y[j, i]:=X[i, j];
End; {Kết thúc thủ tục chuyển vị}
(*=====*)
Begin {Chương trình chính}
    Clrscr;
    Write('Số hàng của ma trận A:'); Readln(m1);
    Write('Số cột của ma trận A:'); Readln(n1);
    Writeln('Nhập ma trận A');
    For i:=1 to m1 do
        For j:=1 to n1 do
            Begin
                Write('A[', i, ', ', j, ']='); Readln(A[i, j]);
            End;
    ChuyenVi(A1, m1, n1, AT1);
    Writeln('In ma trận chuyển vị của ma trận A:');
    For i:=1 to n1 do
        Begin
            For j:=1 to m1 do
                Write(AT1[i, j]:6:2); Writeln;
            End;
        Readln;
    End.

```

6. Trong phần khai báo của chương trình chính ta khai báo:

Type MT=array[1..15, 1..15] of Real;

a. Procedure T2(A: MT; Var T1: Real); *{thủ tục tính tổng bình phương các số hạng nằm trên đường chéo chính của ma trận A và gán vào biến T1}*

Var i: byte;

Tong: Real;

Begin

Tong:=0;

For i:=1 to 15 do Tong:=Tong + A[i,i]*A[i, i];

T1:= Tong;

End;

b. Procedure T3(A: MT; Var T: Real); *{thủ tục tính tổng lập phương các số hạng nằm trên các hàng lẻ và ghi vào biến T}*

Var j, i: byte;

tong: Real;

Begin

Tong:=0; j:=1;

Repeat

For i:=1 to 15 do Tong:=Tong+sq(A[j, i]) *A[j, i];

j:=j+2;

Until j>15;

T:=Tong;

End;

7. Trong phần khai báo của chương trình chính ta khai báo:

Type MT=array[1..15, 1..15] of Real;

```

Procedure T(A: MT; m, n: byte); {tính tổng của các số
                                {hạng âm của ma trận A có m hàng và n cột}
Var i, j: byte;
    TS: Real;
Begin
    TS:=0;
    For i:=1 to m do
        For j:=1 to n do
            If A[i, j] < 0 then TS:=TS+A[i, j];
        Writeln('Tổng của các số hạng âm của ma trận A'
            +'là:', TS);
    End;

```

8. Khai báo trong chương trình chính:

```

Type MT=array[1..100, 1..100] of Real;
Procedure Input(Var A: MT; Ten: Char; m, n: byte);
    {thủ tục nhập ma trận A có m hàng và n cột}
Begin
    For i:=1 to m1 do
        For j:=1 to n1 do
            Begin
                Write(Ten, '[', i, ',', j, ']='); Readln(A[i, j]);
            End;
    End;

```

9. Procedure Change(n: integer; Var St: String);

*{thủ tục chuyển số tự nhiên n cho trước sang
hệ cơ số 2 và được lưu ở trong xâu St}*

```

Const b: array[0..1] of Char = ('0', '1');
Var    du, so: integer; s: string;

```

```

Begin
  S:="";      {xấu rỗng}
  S0:=n;
  Repeat
    Du:=So mod 2;
    So:=So div 2;
    S:=b[du]4S;
  Until (S0=0);
  St:=S;
End;

```

10. Procedure GHPB1(a, b: Real);

```

Begin
  If a=0 then
    Begin
      If b<>0 then
        Write('Phương trình', a, 'x+', b, '=0 vô nghiệm')
      Else Write('Phương trình', a, 'x+', b, '=0 có vô số'
        + 'ng nghiệm x tùy ý');
    End
  Else Write('Phương trình', a, 'x+', b, '=0 có nghiệm'
    + 'là', -b/a);
End;

```

11. Program P4211;

```

Uses Crt;
Var a, b, c, x1, x2: real;
(*=====*)
Procedure Nhapabc;(var aa,bb,cc: real);
Begin

```

```
Write('a='); Readln(aa);
Write('b='); Readln(bb);
Write('c='); Readln(cc);
End;
(*=====*)
Procedure GPTB2;
Var Delta: real;
Begin
  Delta:=sqr(b)-4*a*c;
  If c<>0 then Writeln('Phương trình vô nghiệm')
  Else
    if Delta=0 then
      Begin
        Write('Phương trình có nghiệm kép')
        Write ('x1,2=',-b/(2*a):8:2);
      End
    Else
      Begin
        x1:=(-b+sqrt(Delta))/(2*a);
        x2:=(-b-sqrt(Delta))/(2*a);
        Writeln('Phương trình có hai nghiệm'
          +'phân biệt là');
        Writeln('X1=',x1:8:2, 'X2=',x2:8:2);
      End;
  End;
End;
(*=====*)
Begin { Main Program }
  Clrscr;
```

```

Writeln('Giải phương trình bậc hai với các hệ số:');
Nhapabc(a, b, c);
If a<>0 then GPTB2
Else Writeln('Không phải là phương trình bậc 2');
Readln;
End.

```

12. Được.

13. Program Giai_hpttt;

```

Uses Crt;
Var a1, b1, c1, a2, b2, c2: Real;
(*=====*)
Procedure GHPT (ap1, bp1, cp1, ap2, bp2, cp2: Real);
Var D, Dx, Dy: Real;
Begin
    D:=ap1*bp2 - ap2*bp1;
    Dx:= cp1*bp2 - cp2*bp1;
    Dy:= ap1*cp2 - ap2*cp1;
    If D=0 then Write('Hệ phương trình vô nghiệm'
                    + ' hoặc vô số nghiệm')
    Else Write('Hệ phương trình có nghiệm'
              + 'X=', Dx/D, 'Y=', Dy/D);
End;
(*=====*)
Begin {chương trình chính}
    Clrscr;
    Writeln('Giải Hệ');
    Writeln('a1x+b1y=c1');
    Writeln('a2x+b2y=c2');

```

```

        Writeln('với a1, b1, c1='); Readln(a1, b1, c1);
        Writeln('với a2, b2, c2='); Readln(a2, b2, c2);
        GHPT(a1, b1, c1, a2, b2, c2);
        Readln;
    End.

```

- 14.** Procedure INSERT(St1: string; Var St2: String; Vitri: byte);
 {chèn xâu St1 vào St2 bắt đầu từ vị trí VITRI}

```

    Var i: byte;
        S: String;
    Begin
        If (vitri>length(St2)) or (vitri<1) then
            Write('Không thể chèn ra ngoài xâu')
        Else
            if vitri = 1 then S= St1+St2
            Else
                Begin
                    S:=""; {xâu rỗng}
                    For i:=1 to vitri-1 do S:=S+St2[i];
                    S:=S+St1;
                    For i:=Vitri to length(St2) do S:=S+St2[i];
                    St2:=S;
                End;
            End;
    End;

```

- 15.** Procedure DELETE(Var St: String; Vitri, soky: byte);
 {xóa đi số kí tự là soky kể từ vị trí Vitri trong xâu st}

```

    Var i: byte;
        S: String;

```



```

Begin
  If (vitri<1) or (vitri>length(st)) then
    Write('Không thể xóa ở ngoài chuỗi', St)
  Else
    If sokytu>(length(st) - vitri+1) then
      Write('Không thể xóa được quá số ký tự')
    Else
      Begin
        S:="";
        if vitri = 1 then
          S:= copy(St,Sokytu+1, length(St)-Sokytu)
        Else Begin
          For i:=1 to vitri-1 do S:=S+St[i];
          For i:=Vitri+sokytu to length(St) do S:=S+St[i];
        End
        St:=S;
      End;
End;

```

16. Program P4216;
- ```

Uses Crt;
Var a, b, c: real;
(*=====*)
Procedure Nhap(Var a, b, c: real);
 Procedure input (Var a: real; tenbien: Char);
 Begin
 Repeat
 Write('Nhập' + tenbien+'='); Readln(x1);

```

```

 Until (x1>=0);
End;
Begin {bắt đầu thủ tục nhập}
 Input(a, 'a');
 Input(b, 'b');
 Input(c, 'c');
End; {kết thúc thủ tục nhập}
(*=====*)
Procedure Kiemtra(a, b, c: Real);
Begin
 If (a<b+c) and (b<a+c) and (c<a+b) then
 Write(a, ' ', b, ' và ', c, ' Lập thành ba cạnh của một
 + 'tam giác')
 Else Write('Không lập thành ba cạnh của một tam giác');
End;
(*=====*)
Procedure Trung_tuyen (a, b, c: Real);
Var ma, mb, mc: real;
Begin
 ma:=sqrt((2*sqr(b)+2*sqr(c)-sqr(a))/4);
 mb:=sqrt((2*sqr(a)+2*sqr(c)-sqr(b))/4);
 mc:=sqrt((2*sqr(a)+2*sqr(b)-sqr(c))/4);
 Write('Các trung tuyến của tam giác là:');
 Write('ma=', ma, 'mb=', mb, 'mc=', mc);
End;
(*=====*)
Procedure Dientich(a, b, c: real);
Var p, S: real;
Begin
 p:=(a+b+c)/2;

```

```

 S:=sqrt(p*(p-a)*(p-b)*(p-c));
 Write('Diện tích=', S);
End;
(*=====*)
Begin {chương trình chính}
 Clrscr;
 Nhap(a, b, c);
 Kiemtra(a, b, c);
 Dientich(a, b, c);
 Trung_tuyen(a, b, c);
 Readln;
End.

```

#### 17. Program P4217;

```

 Uses Crt;
 Const Max=50;
 Var P: array[1..Max] of integer;
 id: byte;
 (*=====*)
 Procedure sinh(n:byte);
 Var so,u: integer;
 kt: boolean;
 Begin
 id:=1;
 P[id]:=2;
 so:=3;
 kt:=false;
 Repeat u:=2;
 While (u<=sqrt(so)) and (so mod u<>0) do inc(u);
 if u > sqrt(so) then kt:= True

```

```

Else kt:= False;
if kt= True then
 Begin
 kt:= False;
 inc(id);
 P[id]:=so;
 End
inc(so);
Until (id>n+1);
End;
(*=====*)
Begin
 Clrscr;
 sinh(50);
 Writeln('In 50 số nguyên tố đầu tiên');
 For id:=1 to 50 do
 Begin
 Write(P[id]:5);
 If (id mod 10)=0 then Writeln;
 End;
 Readln;
End.

```

- 18.** Program P4218;  
 Uses Crt;  
 Var OK: Char;  
 a,b,c: real;  
 thoat: boolean;

```

(*=====*)
Procedure Nhapdulieu (Var a1, b1, c1: Real);
Begin
 Writeln('Nhập dữ liệu cho các biến a, b, c:');
 Write('a=');Read(a);
 Write('b=');Read(b);
 Write('c=');Read(c);
End;
(*=====*)
Procedure Giai_pt(ap, bp, cp: Real);
Var Delta, x1, x2: real;
Begin
 Delta:=Sqrt(bp)-4*ap*cp;
 If delta>0 then Write('Pt vô nghiệm')
 Else
 If delta=0 then Write('Pt có nghiệm kép x1, 2=','b/(2*a)');
 Else { a<>0 }
 Begin
 x1:=(-b-sqrt(delta))/(2*a);
 x2:=(-b+sqrt(delta))/(2*a);
 Write('Pt có hai nghiệm tách biệt là:');
 Write('X1=', x1, 'và x2=', x2);
 End;
End;
(*=====*)
Begin {chương trình chính}
 Clrscr;
 thoat:=false;
 While Not thoat do

```

```

Begin
 Clrscr;
 Writeln('1. Nhập dữ liệu');
 Writeln('2. Giải phương trình $ax^2+bx+c=0$ ');
 Writeln('3. Thoát khỏi chương trình');
 Writeln('Chọn (1, 2 hoặc 3):');
 OK:=Readkey;
 Case OK of
 '1': Nhapdulieu(a, b, c);
 '2': Giai_pt(a, b, c);
 '3': Thoat:=True;
 End;
End;
End.

```

**19. Program P4219;**

```

Uses Crt;
Var a, b, c, d, min1, max1: real;
(*=====*)
Procedure Nhap(Var a1, b1, c1, d1: real);
Begin
 Writeln('Nhập dữ liệu cho các biến');
 Writeln('a, b, c, d ='); Readln(a1, b1, c1, d1);
End;
(*=====*)
Procedure Min(a1, b1, c1, d1; Var Mins: Real);
Begin
 Mins:=a1;

```

```
 If Mins>b1 then mins:=b1;
 If Mins>c1 then mins:=c1;
 If Mins>d1 then mins:=d1;
End;
(*=====*)
Procedure Max(a1, b1, c1, d1: real; Var Maxs: Real);
Begin
 Maxs:=a1;
 If Maxs<b1 then maxs:=b1;
 If Maxs<c1 then maxs:=c1;
 If Maxs<d1 then maxs:=d1;
End;
(*=====*)
Begin {chương trình chính}
 Clrscr;
 Nhap(a, b, c,d);
 Min(a, b, c, d, Min1);
 Max(a, b, c, d, Max1);
 Writeln('Min(a, b, c, d)=' , Min1);
 Writeln('Max(a, b, c, d)=' , Max1);
 Readln;
End.
```

20. Program P4220;

Uses Crt;

Type uoc\_nguyen\_to=array[1..50] of longint;

Var u, N: longint;

i, dem: integer;

```

a: uoc_nguyen_to;
(*=====*)
Procedure nhap(Var NN:longint);
Begin
 Repeat
 Write('Nhap N=');Readln(NN);
 Until NN>=0;
End;
(*=====*)
Procedure viet;
Begin
 If dem=0 then
 Writeln('Số ',N,' không thể phân tách được thành',
 +'tích của các số nguyên tố')
 Else
 If dem=1 then Writeln(N,'=',a[dem])
 Else
 Begin
 Write(N,'=');
 For i:=1 to dem-1 do Write(a[i],'*');
 Writeln(a[dem]);
 End;
End;
(*=====*)
Procedure phantich(N1:longint);
Begin
 If N1>=1 then
 Begin

```



```

 u:=2;
 dem:=0;
 Repeat
 If (N1 mod u=0) then
 Begin
 inc (dem);
 a[dem]:=u;
 N1:=N1 div u;
 End
 Else inc(u);
 Until N1=1;
 End
Else dem:=0;
Viet;
End;
(*=====*)
• Begin { Main Program }
 Clrscr;
 Writeln('Phân tích số N thành tích của các số
 +nguyên tố');
 nhap(N);
 phantich(N);
 Write('Nhấn Enter để thoát...');
 Readln;
End.

```

**21. Program P4221;**  
**Uses Crt;**  
**Var N, q: longint;**

```

 st: string;
 (*=====*)
 Procedure Nhap(t: Char; Var Ns: longint);
 Begin
 Repeat
 Write('Nhập '+t+', '=); Readln(Ns);
 If Ns<0 then Writeln('Nhập lại '+t+'>=0');
 Until Ns>=0;
 End;
 (*=====*)
 Function Bieudien(Ns,qs: longint): string;
 {Trả về biểu diễn cơ số Ns theo số ps }
 Var Ng, du: longint;
 ss, st: string;
 Begin
 Ng:=Ns;
 ss:="";
 Repeat
 Du:=Ng mod qs;
 Str(Du,st);
 ss:=st+ss;
 Ng:=Ng div q;
 Until Ng=0;
 Bieudien:=ss;
 End;
 (*=====*)
 Begin { Main Program }
 Clrscr;

```

```
Nhap('N=',N);
Nhap('q=',q);
WriteLn('Biểu diễn ',N,' theo cơ số ',q,' là ',Bieudien(N,q));
Write('Nhấn Enter để thoát...');
ReadLn;
End.
```

### §3. THAM SỐ CỦA THỦ TỤC

1. Kết quả trên màn hình là:

Hãy nhập x: 4

$$x - 1 = 3$$

Giá trị X lúc này bằng 3

2. Kết quả trên màn hình là:

Hãy nhập x: 13

$$X+2 = 2$$

Giá trị của X lúc này bằng 13

3. Kết quả trên màn hình là:

Hãy nhập X: 15

$$X+1 = 16$$

Giá trị của X lúc này bằng 15

4. Kết quả trên màn hình là:

Hãy nhập a: 10

Hãy nhập b: -8

a nhân 2 = -16

Giá trị của a lúc này bằng 10

Giá trị của b lúc này bằng -16

*Giải thích:* Ta nhập giá trị cho 2 biến toàn bộ là a=10; b=-8. Khi gọi thủ tục nhan(b) thì màn hình in ra nội dung:

a nhân 2 = -16

Giá trị của biến b của chương trình chính bây giờ là: -16.

Giá trị của biến a của chương trình chính vẫn không thay đổi và bằng 10.

5. Kết quả trên màn hình:

Cho biết tử số: 100

Cho biết mẫu số: 200

200.0/100.0 = 2.0

6. Program P4306;

Uses: Crt;

Var X, Y, Z: byte;

Begin

Clrscr;

Write('Giải phương trình  $X+Y+X=10$  trong phạm vi  
+ 'số nguyên không âm:');

Writeln('Phương trình có nghiệm là:');

Writeln('x:', 10, 'y:', 10, 'z:', 10);

i:=4;

For X:=0 to 10 do

For Y:=0 to 10-X do

Begin

```
 Z:=10-X-Y;
 Writeln(x: 10, y: 10, z:10);
 inc(i);
 if i=24 then
 Begin
 Write('Nhấn Enter tiếp tục...'); Readln;
 i:=0;
 End
 End
 Write('Nhấn Enter kết thúc...');
 Readln;
End.
```

7. Program P4307;
- ```
Uses Crt;
Var X, Y, Z: byte;
Begin
  Clrscr;
  Writeln('Giải phương trình  $X+Y+Z=12$  trong phạm vi
    '+ 'số nguyên không âm với điều kiện  $x<4$ ');
  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=', X, 'y=', Y, 'z=', Z);
      Readln;
    End.
  End.
```
8. Xem P4309, P4306.
9. Program P4309;
- ```
Uses Crt;
```

```
Var N, a, b, c, X, Y, Z, i: Integer;
Begin
 Clrscr;
 Write(' N, a, b, c = '); Readln(N, a, b, c);
 if (a+b+c-3<N) then
 Begin
 Write('Phương trình vô nghiệm!'); Readln;
 End
 Else
 Begin
 Writeln('Phương trình có nghiệm là:');
 Writeln('x': 10, 'y': 10, 'z':10);
 i:=4;
 For X:=0 to (a-1) do
 For Y:=0 to (b-1) do
 For Z:=0 to (c-1) do
 If (X+Y+Z=N) then
 Begin
 Writeln(x: 10, y: 10, z:10);
 inc(i);
 if i=24 then
 Begin
 Write('Nhấn Enter tiếp tục...'); Readln;
 i:=0;
 End;
 End;
 End;
 End;
 End;
 End;
 Write('Nhấn Enter kết thúc...');
```

```
 Readln;
End.
```

```
10. Program P4310;
 Uses Crt;
 Var an, a1: real;
 k: integer;
 Begin
 Clrscr; '
 a1:=2;
 For k:=1 to 9999 do
 Begin
 an:=a1*(6-sqr(a1))/4;
 a1:=an;
 End;
 Write(an);
 Readln;
 End.
```

Nhận xét: dãy số này tiến tới căn bậc hai của 2.

```
11. Program P4311;
 Uses Crt;
 Var st: string;
 k: byte;
 (*=====*)
 Procedure nhapxau(Var xau: string);
 Var ch: Char;
 Begin
 xau:=""; {Xâu rỗng}
```

```

Repeat
 ch:=Readkey;
 If ch in ['0','1'] then
 Begin
 Write(ch);
 xau:=concat(xau,ch);
 End;
Until ch=#13;
End;

(*=====*)
Function check(XAU: string; k1: byte): boolean;
{Hàm trả về giá trị TRUE nếu xâu kí tự XAU lặp bậc K1
trường hợp còn lại hàm trả về giá trị FALSE}
Var stop,kt: boolean;
 len, i, j: byte;
 s1, s2: string;
Begin
 stop:=false;
 len:=length(XAU);
 i:=1;
 While (not stop)and(i+k1-1<=len) do
 Begin
 s1:=copy(XAU,i,k1);
 j:=i+k1; kt:=false;
 While (j+k1-1<=len)and(not kt) do
 Begin
 s2:=copy(XAU,j,k1);
 If s2=s1 then kt:=true

```



```

 Else inc(j);
 End;
 If (kt=true)or(j+k1-1>len) then stop:=true
 Else inc(i);
 End;
 check:=kt;
End;
(*=====*)
Begin
 Clrscr;
 Write('Nhập xâu kí tự nhị phân:');
 nhapxau(st);
 Write('Bậc lặp :');Readln(k);
 If check(st,k)=true then
 Write('Xâu nhị phân ',st,' lặp bậc ',k)
 Else Write('Xâu nhị phân ',st,' không lặp bậc ',k);
 Readln;
End.

```

## 12. Program P4312;

```

Uses Crt;
Var st: string;
(*=====*)
Procedure nhapxau(Var S: string);
Var ch: Char;
Begin
 S:=""; {Xâu rỗng}
 Repeat

```

```

ch:=Readkey;
If ch in ['0','1'] then
 Begin
 Write(ch); S:=concat(S,ch);
 End;
Until ch=#13;
Writeln;
End;
(*=====*)
Function BL(SS: string): byte;
{Hàm trả về bậc không lặp của xâu nhị phânSS}
Var stop, kt: boolean;
 len, i, j, k: byte;
 s1, s2: string;
Begin
 stop:=false;
 len:=length(SS);
 i:=1;
 k:=2;
 While (not stop)and(i+k-1<=len) do
 Begin
 s1:=copy(SS,i,k);
 j:=i+k;
 kt:=false;
 while (j+k-1<=len)and(not kt) do
 Begin
 s2:=copy(SS,j,k);
 If s2=s1 then kt:=true

```

```

 Else inc(j);
 End;
 If kt=false then
 If (i=len-2*k+1) then stop:=true
 Else inc(i)
 Else
 Begin
 i:=1;inc(k);
 End;
 End; { While}
 BL:=k;
 End;
 (*=====*)
 Begin
 Clrscr;
 Write('Nhập xâu nhị phân: ');
 nhapxau(st);
 Write('Xâu nhị phân ',st,' không lặp bậc ', bl(st));
 Readln;
 End.

```

### 13. Program P4313;

```

 Uses Crt;
 Var a: array[1..5] of 0..1;
 (*=====*)
 Procedure print(np: byte);
 Var ip: byte;
 Begin

```

```
For ip:=1 to np do Write(a[ip]);
Writeln;
End;
(*=====*)
Function check(nc: byte): boolean;
Var k: byte;
 kt: boolean;
Begin
 k:=1;
 kt:=false;
 While not kt and (k<=nc-1) do
 Begin
 kt:=(a[k]=0)and(a[k+1]=0);
 inc(k);
 End;
 check:=kt;
End;
(*=====*)
Procedure list (n: byte);
Var thoat, kt: boolean;
 dem, i, j: byte;
Begin
 Writeln('Liệt kê các chuỗi nhị phân "Chuẩn" độ dài ',n);
 For i:=1 to n do a[i]:=0;
 thoat:=false;
 dem:=0;
 While not thoat do
 Begin
```

```
 If not check(n) then
 Begin
 inc(dem);
 print(n);
 End;
 i:=n;
 While (i>=1)and(a[i]=1) do
 Begin
 a[i]:=0;
 dec(i);
 End;
 If i>=1 then a[i]:=1
 Else thoát:=true;
End;
Writeln('Có tất cả ',dem,' xâu nhị phân chuẩn có
 '+ 'độ dài là',n);
Write('Nhấn phím bất kì để thoát hoặc tiếp tục...');
Readkey;
Clrscr;
End;
(*=====*)
Begin { Main Program }
 Clrscr;
 list(3);
 list(4);
 list(5);
End.
```

d.  $A_1=2; A_2=3; A_n=A_{n-1}+A_{n-2} (n \geq 3)$  (Xem bài 14).

14. Chia các xâu nhị phân chuẩn độ dài  $n (n \geq 3)$  làm hai loại:

- Loại 1 có kí tự cuối cùng là số '1'.

- Loại 2 có kí tự cuối cùng là số '0'.

Ta nhận xét rằng:

Nếu bỏ kí tự cuối cùng đi thì các xâu loại 1 trở thành một xâu nhị phân chuẩn độ dài  $n-1$ . Ngược lại, từ một xâu nhị phân chuẩn bất kì độ dài  $n-1$ , ta có thể thêm kí tự '1' vào cuối để thu được các xâu loại 1. Như vậy số các xâu loại 1 là  $A_{n-1}$ .

Các xâu loại 2 có kí tự kế kí tự cuối cùng là '1'. Nếu bỏ kí tự đó cùng kí tự cuối cùng đi, ta sẽ thu được một xâu nhị phân chuẩn độ dài  $n-2$ . Ngược lại, từ một xâu nhị phân chuẩn độ dài  $n-2$  bất kì, ta có thể thêm '10' vào cuối cùng để nhận được một xâu loại 2. Như vậy số các xâu loại 2 là  $A_{n-2}$ .

Tóm lại ta có  $A_n = A_{n-1} + A_{n-2}$  - Điều phải chứng minh.

Minh họa:                      Số lượng

Các xâu loại 1:      Một xâu nhị phân chuẩn độ dài  $n-1$  + '1' ( $A_{n-1}$ )

Các xâu loại 2:      Một xâu nhị phân chuẩn độ dài  $n-2$  + '01' ( $A_{n-2}$ )

Tổng cộng:       $A_n = A_{n-1} + A_{n-2}$

15. Program P4315;

Uses Crt;

Var Bit: array[1..10] of '0'..'1';

(\*-----\*)

Procedure print(np, d: byte);

```

Var ip: byte;
Begin
 if d mod 10=0 then
 Begin
 Write('Nhấn một phím bất kì để tiếp tục...');
 Readkey;
 End;
 For ip:=1 to np do Write(Bit[ip]);
 Writeln;
End;

(*=====*)
Function check(nc: byte): boolean;
 {Hàm trả về giá trị TRUE nếu xâu nhị phân có độ dài nc
 không chứa 3 kí tự 0 liên tiếp; trường hợp còn lại hàm trả
 về giá trị FALSE}
 Var k: byte;
 kt: boolean;
 Begin
 k:=1;
 kt:=false;
 while not kt and (k<=nc-2) do
 Begin
 kt:=(Bit[k]='0')and(Bit[k+1]='0')and(Bit[k+2]='0');
 inc(k);
 End;
 check:=not kt;
 End;

(*=====*)

```

```

Procedure list(n: byte);
Var thoat, kt: boolean;
 dem, i, j: byte;
Begin
 Writeln('Liệt kê các xâu nhị phân có độ dài 6');
 Writeln('và không chứa ba kí tự 0 liên tiếp');
 For i:=1 to n do Bit[i]:='0';
 thoat:=false;
 dem:=0;
 While not thoat do
 Begin
 If check(n) then
 Begin
 inc(dem); print(n,dem);
 End;
 i:=n;
 While (i>=1)and(Bit[i]='1') do
 Begin
 Bit[i]:='0'; dec(i);
 End;
 If i>=1 then Bit[i]:='1'
 Else thoat:=true;
 End;
 Writeln('Có tất cả ',dem,' xâu như vậy');
End;

(*=====*)
Begin { Main Program }
 Clrscr;

```



```

list(6);
Write('Nhấn Enter để thoát...');
Readln;
End.

```

16. Tương tự bài 15, ta sẽ chia các xâu nhị phân độ dài  $n$  và không chứa 3 số 0 liên tiếp làm 3 loại:

- Loại 1 có kí tự cuối cùng là '1'.
- Loại 2 có kí tự cuối cùng là '0', kí tự kế cuối cùng là '1'.
- Loại 3 có kí tự cuối cùng là '0', kí tự kế cuối cùng là '0'.

Nhận xét tương tự bài 14 ta sẽ được:

- Số các xâu loại 1 là  $B_{n-1}$ .
- Số các xâu loại 2 là  $B_{n-2}$ .
- Số các xâu loại 3 là  $B_{n-3}$ .

Như vậy ta có công thức  $B_n = B_{n-1} + B_{n-2} + B_{n-3}$  với mọi  $n \geq 4$ .

$$B_1=2, B_2=4, B_3=7.$$

Để tách  $A_n$  ta nhận xét rằng:

$$A_{n-3} = 2^{n-3} - B_{n-3}$$

$$A_{n-2} = 2^{n-2} - B_{n-2}$$

$$A_{n-1} = 2^{n-1} - B_{n-1}$$

$$A_{n-1} + A_{n-2} + A_{n-3} = (2^{n-1} + 2^{n-2} + 2^{n-3}) - (B_{n-1} + B_{n-2} + B_{n-3}) = 7 \cdot 2^{n-3} - B_n$$

Nhưng vì  $B_n = 2^n - A_n$  nên

$$A_{n-1} + A_{n-2} + A_{n-3} = 7 \cdot 2^{n-3} - (2^n - A_n) = -2^{n-3} + A_n$$

$$\Rightarrow A_n = A_{n-1} + A_{n-2} + A_{n-3} + 2^{n-3} \text{ với mọi } n \geq 4.$$

$A_1 = 0; A_2 = 0, A_3 = 1.$

17. Program P4317;

Const Maxp=1000;

Var Stack: array[1..maxp] of integer;

p, v: integer; {con trỏ trỏ vào phần tử đưa vào cuối cùng}

(\*=====\*)

Procedure Push (v: integer); {đưa phần tử v vào stack}

Begin

inc(p);

stack[p]:=v;

End;

(\*=====\*)

Procedure Pull (Var v: integer); {lấy một phần tử khỏi stack và lưu vào v}

Begin

If p>0 then

Begin v:=stack[p]; dec(p); End

Else Write('Stack đã hết phần tử');

End;

(\*=====\*)

Begin

Clrscr;

p:=0;

Write('Đưa số 5 vào stack');

Push(5);

Write('Lấy ra 1 phần tử khỏi Stack');

Pull(v);

```

 Write(v);
 Readln;
End.

```

18. Program P4318;
- ```

Uses Crt;
Const maxp=100;
Var Queue: array[1..maxp] of integer;
    v, p: integer;
(*=====*)
Procedure push(v: integer); {đưa phần tử v vào hàng đợi
                             Queue}
Var i: integer;
Begin
    inc(p);
    if p = 1 then stack[p]:=v
    Else
        Begin
            For i:=p downto 2 do
                stack[i]:=stack[i-1]; stack[1]:=v;
        End;
End;
(*=====*)
Procedure Pull (Var v: integer); {lấy một phần tử
                                 khỏi Queue}
Begin
    If p<max then
        Begin
            v:=Queue[p];

```

```

End
Else Write('Queue đã rỗng');
End;
(*=====*)
Begin {chương trình chính}
  Clrscr;
  Max:=0; p:=0;
  Writeln('Đưa số 5 vào Queue');
  Push(5);
  Writeln('Đưa số -1 vào Queue');
  Push(-1);
  Write('Lấy 1 phần tử khỏi Queue');
  Pull(v);
  Write('Phần tử đó là', v);
  Readln;
End.

```

19. Nguồn gốc số Fibonacci:

Gọi số thỏ trên đảo sau k tháng là T_k . Rõ ràng $T_1=1=F_1$,
 $T_2=1=F_2$, $T_3=2=F_3$.

Ta chia số thỏ thành 3 loại: a_k - trưởng thành, b_k -lớn (từ 1 đến dưới 2 tháng) và c_k -nhỏ (mới sinh đến dưới 1 tháng).

$$\begin{aligned}
 \text{Như vậy } T_k &= a_k + b_k + c_k \\
 T_{k-1} &= a_{k-1} + b_{k-1} + c_{k-1} \\
 T_{k-2} &= a_{k-2} + b_{k-2} + c_{k-2}
 \end{aligned}$$

Ta có các công thức liên hệ sau:

$$a_k = a_{k-1} + b_{k-1}$$

$a_{k-1} = a_{k-2} + b_{k-2}$ (vì sau một tháng số thỏ lớn sẽ trưởng thành)

$$b_k = c_{k-1}$$

$b_{k-1} = c_{k-2}$ (vì sau một tháng số thỏ nhỏ sẽ lớn)

$c_k = a_k = a_{k-1} + b_{k-1}$ (vì mỗi thỏ trưởng thành sẽ sinh ra một thỏ con).

$$\begin{aligned} \text{Như vậy } T_k &= a_k + b_k + c_k = (a_{k-1} + b_{k-1}) + c_{k-1} + (a_{k-1} + b_{k-1}) \\ &= (a_{k-1} + b_{k-1} + c_{k-1}) + [(a_{k-2} + b_{k-2}) + c_{k-2}] \\ &= T_{k-1} + T_{k-2}. \end{aligned}$$

Vậy số cặp thỏ trên đảo chính bằng số Fibonacci - Điều phải chứng minh.

§4. KHÁI NIỆM HÀM

1. Program P4401;

Uses Crt;

Var x: Real;

(*=====*)

Function f (x: Real): Real;

Begin

f:=sqr(x)*x-3*sqr(x)+1;

End;

(*=====*)

Begin {Main Program}

Clrscr;

Write ('Nhập x='); Readln (x);

```
Write ('f(x) = x3-3x2+1=', f(x));
```

```
Readln;
```

```
End.
```

2. Program P4402;

```
Uses Crt;
```

```
Var a: Real;
```

```
    N: Integer;
```

```
(*=====*)
```

```
Function a_mu_n (a1: Real; n1: integer): Real;
```

```
Var i: integer;
```

```
    T: Real;
```

```
Begin
```

```
    T:=a1;
```

```
    For i:=2 to n1 do T:=a1*T;
```

```
    a_mu_n:=T;
```

```
End;
```

```
(*=====*)
```

```
Begin {Main Program }
```

```
    Clrscr;
```

```
    Write ('Nhập a, n='); Readln(a, n);
```

```
    Write (' a mũ n =', a_mu_n (a,n));
```

```
    Readln;
```

```
End.
```

3. Function Check (n: integer): Boolean;

{nếu n nguyên tố thì hàm trả về giá trị True; nếu không
thì hàm trả về giá trị False}

```
Var KT: Boolean;
```

```
      i: Integer;
Begin
    KT:=True;
    i:=2;
    While (i<=n div 2) and (KT) do
        If (n mod i = 0) then KT:=False
        Else inc(i);
    Check:= KT;
End;
```

4. Program P4404

```
Uses Crt;
Var Na, Nb: integer;
(*=====*)
Function USCLN(a, b: integer): integer;
Var r: integer;
Begin
    While b>0 do
        Begin
            r:=a mod b; a:=b; b:=r;
        End;
    USCLN:=a;
End;
(*=====*)
Begin
    Clrscr;
    Write('a,b='); Readln(a,b);
    Write('USCLN('a,',',b,')=',USCLN(a,b));
End.
```

5. Function tach_tu(St: String): String; *{trả về từ đầu tiên của xâu kí tự St}*

```
Var s: string;  
    i: byte;  
Begin  
    i:=1; S:= ""; {xâu rỗng}  
    While (i<=length (St)) and (St[i]<>#32) do  
        Begin  
            S:=S+St[i]; Inc(i);  
        End;  
    Tach_tu:=S;  
End;
```

6. Function check (St: String): Boolean;

{nếu xâu St đối xứng thì hàm check nhận giá trị True còn nếu không thì hàm check nhận giá trị False}

```
Var KT: Boolean;  
    i, len: byte;  
Begin  
    len:= length (St);  
    KT:=True;  
    i:=1;  
    While (i<=len div 2) and (KT) do  
        If St[i]<>st[len-i+1] then KT:= False  
        Else inc(i);  
    Check:= KT;  
End;
```

7. Program P4407;
Uses Crt;


```

Var X, Y: Real;
(*=====*)
Function X_mu_Y (X1,Y1: Real): Real;
Begin
    X_mu_Y:=exp(y*ln(x));
End;
(*=====*)
Begin    {Main program}
    Clrscr;
    Write ('Nhập x, y='); Readln (X, Y);
    Write ('x mũ y=', x_mu_y (x,y));
    Readln;
End.

```

8. Program P4408;

```

Uses Crt;
Type Mang= array [1..1000] of integer;
Var a: Mang; N: integer;
(*=====*)
Function USCLN (b: Mang; num: integer): integer;
    {tính ước số chung lớn nhất của (b1, b2, ...bnum)}
Var ap, bp, r: integer;
Begin
    ap:=b[1];
    i:= 2;
    Repeat
        bp:=b[i];
        While bp>0 do

```

```

        Begin
            r:= ap mod bp;
            ap := bp;
            bp := r;
        End;
    inc(i);
    Until (i>num);
    USCLN:=ap;
End;
(*=====*)
Begin {Chương trình chính}
    Clrscr;
    Write('N ='); Readln(N);
    For i:=1 to N do
        Begin
            Write('a', i, '='); Readln(a[i]);
        End
    Write('USCLN a1, a2, ... aN=', USCLN(a, N));
End.

```

9. Function Int (x: Real): integer;
 {trả về số nguyên lớn nhất không vượt quá X}
 Begin
 Int:= Round (x-0,5);
 End;
10. Function SN_gan_x(x: Read):integer;
 {Tính số nguyên gần số thực X nhất}
 Begin

```
Sn_gan_X:= Round(X);
```

```
End;
```

11. Function KC(x: Read): integer;

{ trả về khoảng cách gần nhất của số thực X với các số nguyên: ví dụ KC (5.125) = 0.125, KC (10.625)=0.375}

```
Begin
```

```
    KC:= ABS(X- round (X));
```

```
End;
```

12. Function PHI (n: integer): integer;

{ Trả về số các các số tự nhiên nhỏ hơn n và nguyên tố cùng nhau với n}

```
Var i, Dem: Integer;
```

```
(*=====*)
```

```
Function USCLN (a, b: integer): integer;
```

```
Var r: integer; {tìm USCLN của a và b}
```

```
Begin
```

```
    While b>0 do
```

```
        Begin
```

```
            r:=a mod b; a:=b; b:=r;
```

```
        End;
```

```
    USCLN:= a;
```

```
End;
```

```
(*=====*)
```

```
Begin { bắt đầu hàm PHI}
```

```
    Dem:= 0;
```

```
    For i:=1 to (n-1) do
```

```
        If USCLN (i,n)=1 then inc (dem);
```

```
    PHI:=Dem;
```

End; { kết thúc hàm PHI}

13. Function Giai_thua1 (n: byte) : longint;

Var GT: longint;

i: byte;

Begin

GT:=1;

For i:=2 to n do GT:=GT*i;

Giai_thua1:=GT;

End;

14. Function Giai_thua2 (n: byte): longint;

Var GT: longint;

i: byte;

Begin

GT:=1;

If (n mod 2 =0) then i:=2

Else i:=1;

While (i<=n) do

Begin

GT:=GT*i;

i:=i+2; {hoặc inc (i,2)}

End;

Giai_thua2:=GT;

End;

15. Function So_vt_khac (S1, S2: String): byte;

Var i, So: byte;

Begin

So:=0;

```
    For i:=1 to length(S1) do
        If S1[i] <> S2[i] then inc(So);
    So_vt_khac:=So;
End;
```

16. Program P4416;

```
Uses Crt;
Var S: String;
    i: byte;
(*=====*)
Function Sph (St: string; i: byte): string;
Var S1: String;
Begin
    S1:=St;
    If S1[i] = '0' then S1[i]:='1'
    Else s1[i]:='0';
    sph:=S1;
End;
(*=====*)
Begin {chương trình chính}
    CLRSCR;
    Write('nhập xâu S='); Readln(S);
    Writeln ('Các xâu nhị phân Sph cùng độ dài với S và
        + 'có d (S, Sph)=1 là: ');
    For i:=1 to length(s) do Writeln(Sph(S,i));
    Readln;
End.
```

```
17. Program P4417;
  Uses Crt;
  Const Max=10000;
  Var  N, i, dem: longint;
        a: array[1..Max] of longint;
  (*=====*)
  Function check(M: longint): boolean;
  Var  stop, kt: boolean;
        kd, bd, T: longint;
  Begin
    stop:=false;
    kt:=false;
    kd:=0;
    While not stop do
      Begin
        bd:=kd;
        dem:=0;
        T:=0;
        If bd>=(M+1) div 2 then stop:=true
        Else
          Begin
            While T<M do
              Begin
                inc(dem);
                T:=T+bd;
                a[dem]:=bd;
                inc(bd);
              End;
```

```
        If T=M then
            Begin
                kt:=true; stop:=true;
            End
            Else inc(kd);
        End;
    End;
    Check:=kt;
End;
(*=====*)
Begin
    Clrscr;
    Repeat
        Write('Nhập số tự nhiên N='); Readln(N);
        If N<0 then Writeln('Nhập lại N>=0');
    Until N>=0;
    If check(N) then
        Begin
            Write(N,'=');
            For i:=1 to dem-1 do Write(a[i],'+');
            Write(a[dem]);
        End
    Else Write('Số ',N,' không thể biểu diễn được thành'
        +tổng của >=2 số tự nhiên liên tiếp');
    Readln;
End.
```

18. Program P4418;
Uses Crt;

```
Var N:integer;
(*=====*)
Function check(M: integer): boolean;
Var dem, so, s: integer;
Begin
    dem:=0;
    so:=2;
    Repeat
        If M mod so=0 then
            Begin
                inc(dem); M:=M div so;
            End
        Else inc(so);
    Until (M=1);
    check:=(dem=3);
End;
(*=====*)
Begin
    Clrscr;
    Writeln('Nhập một số tự nhiên');
    Repeat
        Write('N='); Readln(N);
        If N<=0 then Write('Nhập lại. ');
    Until N>0;
    If check(N) then Write('Số',N,'là tích bậc 3')
    Else Write('Số ',N,'Không phải là tích bậc 3');
    Readln;
End.
```


19. Ta có khai triển nguyên tố của một số là khai triển có số nhân tử cực đại. Vì $N = P_1^{k_1} \cdot P_2^{k_2} \cdot \dots \cdot P_s^{k_s}$ là khai triển nguyên tố, suy ra N chỉ có thể biểu diễn được thành tích của nhiều nhất là $(k_1 + k_2 + \dots + k_s)$ số tự nhiên lớn hơn 1.

Như vậy N là số có tích thực sự bậc $(k_1 + k_2 + \dots + k_s) - \text{đpcm}$.

20. Program P4420;

Uses Crt;

Var N: integer;

(*=====*)

Function Btts(M: integer): integer;

Var d, so: integer;

Begin

d:=0;

so:=2;

Repeat

If M mod so=0 then

Begin

inc(d); M:=M div so;

End

Else inc(so);

Until (M=1);

btts:=d;

End;

(*=====*)

Begin

Clrscr;

Write('Input N='); Readln(N);

```
Write('Bậc tích thực sự của', N, 'là', btt(N));
Readln;
End.
```

21. Program P4421;

```
Uses Crt;
Const so: array['0'..'9'] of byte=(0,1,2,3,4,5,6,7,8,9);
Var U: string;
(*=====*)
Function N(s: string): shortint;
Var i, kq: shortint;
Begin
    kq:=0;
    If length(s)<>0 then
        For i:=1 to length(s) do
            If (so[s[i]] mod 2=0) then inc(kq,1)
            Else inc(kq,-1);
        N:=kq;
    End;
(*=====*)
Procedure nhap(Var st: string);
{Nhập một xâu ký tự bao gồm các kí tự từ 0 đến 9}
Var ch: char;
Begin
    st:=""; { Xâu rỗng }
    Repeat
        ch:=readkey;
        If ch in ['0'..'9'] then
```

```

        Begin
            Write(ch); st:=concat(st,ch);
        End;
    Until ch=#13;
    Writeln;
End;
(*=====*)
Begin
    Clrscr;
    Writeln('N("1233536535"+')= ',N('1233536535'));
    Write('Nhập xâu U=');
    nhap(U);
    Write('N(U)=N("',U,'"')= ', N(U));
    Readln;
End.

```

22. Program P4422;

```

Uses Crt;
Type mang=array [1..100] of integer;
Var i, j, n, d, a1, p1, q: integer;
    a, p: mang;
(*=====*)
Function USCLN(a, b: integer): integer;
Var c, a1, b1: integer;
Begin
    a1:=a;
    b1:=b;
    Repeat
        c:=a1 mod b1;

```

```
        a1:=b1;
        b1:=c;
    Until (c=0);
    uscln:=a1;
End;
(*=====*)
Function US(i: integer): integer;
Var j, d: integer;
Begin
    d:=a[j];
    For j:=i to n do d:=uscln(d,a[j]);
    us:=d;
End;
(*=====*)
Procedure sop(a, b: integer; var p,q: integer);
Var d, i, j: integer;
    ok: boolean;
Begin
    d:=uscln(a,b);
    ok:=false;
    i:=b;
    Repeat
        j:=-a;
        Repeat
            If (i*a+j*b =d) then
                Begin
                    ok:=true; p:=i; q:=j;
                End;
```

```
        inc(j);
    Until (j>a)or(ok);
    inc(i);
    Until (i>b)or(ok);
End;
(*=====*)
Begin
    Clrscr;
    Write('Nhập số n: '); Readln(n);
    Writeln('Nhập các phần tử của mảng a: ');
    For i:=1 to n do
        Begin
            Write('  a',i,' = '); Readln(a[i]);
            p[i]:=1;
        End;
    For i:=1 to n-1 do
        Begin
            a1:=us(i+1);
            sop(a[i],a1,p1,q);
            p[i]:=p[i]*p1;
            For j:=i+1 to n do p[j]:=p[j]*q;
        End;
    For i:=1 to n do Write('p',i,' = ',p[i],'.');
    Readln;
End.
```

§5. BIẾN CỤC BỘ, BIẾN TỔNG THỂ TRUYỀN THAM BIẾN CHO THỦ TỤC / HÀM.

1. a. Đúng.

b. Đúng.

c. Đúng.

2. a. Đúng

b. Sai

c. Sai

3. Có 3 cách truyền giá trị cho biến trong thủ tục và hàm.

Cách 1: truyền bằng biến tổng thể.

Cách 2: truyền bằng đối dạng tham biến giá trị.

Cách 3: truyền bằng đối dạng tham số biến.

4. Sai vì thủ tục TINH sử dụng tham số biến X, Y còn lời gọi thủ tục TINH bên trong chương trình chính lại không sử dụng tham số biến để truyền giá trị.

5. Được

6. Giải thích;

a. Cả 2 tham số x và y của thủ tục VD_thamso đều là tham trị cho nên khi gọi thủ tục VD_thamso (a, b) với a=1; b=4 thì x=2; y=5 và trên màn hình in ra:

2 5

Kết thúc thủ tục này biến a và b không bị thay đổi giá trị vì vậy lệnh *WriteIn* (a:5, b:5); in ra

1 4

b. Thủ tục *VD_thamso* sử dụng tham trị x và tham số biến y cho nên khi gọi thủ tục *VD_thamso(a, b)* thì cũng giống a. màn hình hiện:

2 5

Tuy nhiên khi kết thúc thủ tục biến a vẫn giữ nguyên giá trị, còn biến b bị thay đổi nên màn hình hiện:

1 5

c. Thủ tục *VD_thamso* sử dụng 2 tham số hình thức là tham biến cho nên khi gọi thủ tục *VD_thamso (a, b)* thì màn hình xuất hiện:

2 5

Kết thúc thủ tục này biến a, b bị thay đổi cho nên lệnh *WriteIn(a:5, b:5)* in ra màn hình

2 5

7. Được vì trong Turbo Pascal có từ khóa *Forward* giúp cho ta làm được điều này.

Ví dụ: chương trình tính tổng của 10 số tự nhiên liên tiếp dưới đây: $S=1+2+...+10$.

```

Program Tong;
Uses Crt;
Var i, s: byte;
Procedure P1; Forward;
Procedure P2; Forward;
(*=====*)
Procedure P1;
  Begin

```

```
        If i < 10 then
            Begin
                inc(i);
                S:=S+i;
                P2; {gọi thủ tục P2}
            End;
        End;
    End;
    (*=====*)
    Procedure P2;
    Begin
        If i < 10 then
            Begin
                inc(i);
                S:=S+i;
                P1; {gọi thủ tục P1}
            End;
        End;
    End;
    (*=====*)
    Begin {Main Program}
        Clrscr;
        i:=0;
        s:=0;
        P1;
        Write ('1+2+...10=', S);
        Readln;
    End
```

8. Được (đây chính là thủ tục hoặc hàm đệ quy).


```
9. Program P4509;
   Uses Crt;
   Var a, b, c: integer;
   (*=====*)
   Function uscln(x1, y1: integer): integer;
   Var r: integer;
   Begin
       While y1>0 do
           Begin
               r:=x1 mod y1;
               x1:=y1; y1:=r;
           End;
       uscln:=x1;
   End;
   (*=====*)
   Procedure Giai_pt(a1, b1, c1: integer);
   Var k, x, y: integer;
       stop: boolean;
   Begin
       k:=uscln(uscln(a1,b1),c1);
       a1:=a1 div k;
       b1:=b1 div k;
       c1:=c1 div k;
       Write('Phương trình');
       If uscln(a1,b1)<>1 then Write(' vô nghiệm')
       Else
           Begin
               x:=c1*b1;
```

```

stop:=false;
Repeat
    y:=c1*a1;
    Repeat
        If (a1*x+b1*y=c1) then
            Begin
                Write(' co nghiệm X=',x,' Y=',y);
                stop:=true;
            End;
        dec(y,c1);
    Until (y=-c1*a1) or stop;
    dec(x,c1);
Until (x=-c1*b1) or stop;
End;
End;
(*=====*)
Begin
    Clrscr;
    Writeln('Giải phương trình Ax+By=C trong đó');
    Write('A=');Readln(a);
    Write('B=');Readln(b);
    Write('C=');Readln(c);
    Giai_pt(a,b,c);
    Readln;
End.

```

10. Program P4510;

Uses Crt;

```

Var  tg, p, q, k: longint;
(*=====*)
Procedure input(ten: char; Var STN: longint);
Begin
  Repeat
    Write(ten, '='); Readln(STN);
    If STN<0 then
      Writeln(#7, 'Nhập lại. Đây không phải là số tự nhiên');
  Until STN>=0;
End;
(*=====*)
BEGIN { Main Program }
Clrscr;
Writeln('      1');
Writeln('      1');
Writeln('  a1+  1');
Writeln('      1');
Writeln('  a2   1');
Writeln('      1');
Writeln('      a3 + ... + 1');
Writeln('              an');
Writeln('');
input('p',p);
Repeat
  input('q',q);
  If q<=p then Writeln('Nhập lại. Q phải lớn hơn P');
Until q>p;
k:=0;

```

```

Writeln('Ta có:');
While (p<>0) do
  Begin
    inc(k);
    Write('a[':4,k,']=',q div p);
    tg:=p;
    p:=q mod p;
    q:=tg;
  End;
Writeln;
Write('Nhấn Enter để kết thúc...');
Readln;
END.

```

11. Program P4511;

```

Uses Crt;
Var xau1,xau2,xau: string;
(*=====*)
Procedure compare(s1, s2: string; Var kq: string);
Var i: byte;
(*=====*)
Function kt(ch: char; st: string): boolean;
{Kiểm tra xem kí tự ch có trong xâu st hay không. Nếu có
thì hàm trả về giá trị True. Nếu không thì hàm trả về giá
trị False}
Begin
  kt:=pos(ch,st)<>0;
End;
(*=====*)

```

```

Begin {Thân của thủ tục compare}
  kq:="";      {Xâu rỗng}
  For i:=1 to length(s1) do
    If (not kt(s1[i],kq))and(kt(s1[i],s2)) then
      kq:=concat(kq,s1[i]);
End;
(*=====*)
Begin
  Clrscr;
  Writeln('Nhập 2 xâu kí tự S1 và S2');
  Write('S1 ');Readln(xau1);
  Write('S2 ');Readln(xau2);
  compare(xau1, xau2, xau);
  If xau<>" then Writeln('Xâu chung là:',xau)
  Else Writeln('Không có kí tự nào có trong cả hai'
    +'xâu trên. ');
  Write('Nhấn ENTER để thoát...');
  Readln;
End.

```

12. Program P4512

```

Uses Crt;
Const M=100;
Var  S: array[1..M] of string;
      max, min, i, j, n: byte;
(*=====*)
Function d(U,V: string): byte; {Trả về tổng số loại kí tự
không giống nhau trong 2 xâu U và V}
Var k, id: byte;

```

```

    s, luu: string;
Begin
    luu:=""; {Xâu rỗng}
    For id:=1 to length(U) do
        If (pos(U[id],V)=0)and(pos(U[id],luu)=0) then
            luu:=concat(luu,U[id]);
    For id:=1 to length(V) do
        If (pos(V[id],U)=0)and(pos(V[id],luu)=0) then
            luu:= concat(luu,V[id]);
    d:=length(luu);
End;
(*=====*)
Procedure nhap;
Begin
    Repeat
        Write('Số xâu kí tự(>=2):'); Readln(n);
        If n<2 then
            Writeln('#7, 'Có ',n,' xâu kí tự nên không thể so'
                +'sánh được. Nhập lại');
    Until n>=2;
    Writeln('Nhập ',n,' xâu kí tự');
    For i:=1 to n do
        Begin
            Write('S',i,'='); Readln(S[i]);
        End;
    End;
(*=====*)
Begin { Main Program }

```

```

Clrscr;
nhap;
max:=0;
min:=255;
For i:=1 to n-1 do
  For j:=i+1 to n do
    Begin
      If max<d(S[i],S[j]) then max:=d(S[i],S[j]);
      If min>d(S[i],S[j]) then min:=d(S[i],S[j]);
    End;
  Write('Max(d(Si,Sj)=' ,max,' Min(d(Si,Sj)=' ,min);
Readln;
End.

```

13. Program CT4513;

```

Uses Crt;
Var  Sobonghoa, Motlanhai, Dahai: integer;
     Mayhai: boolean;
     Ch, tt: char;
(*=====*)
Procedure Play;
Begin
  tt:='C';
  While Ucase(tt)='C' do
    Begin
      Repeat
        Write('Cho số bông hoa(>1):'); Readln(Sobonghoa);
        If Sobonghoa<=1 then Write(#7,'Nhập lại.');
```

```

Until Sobonghoa>1;
Repeat
    Write('Một lần hái tối đa là bao nhiêu bông. ');
    Readln(Motlanhai);
    If Motlanhai>=Sobonghoa then
        Write(#7,'Nhập lại. ');
Until (0<Motlanhai)and(Motlanhai<Sobonghoa);
Write('Máy hái trước (M) – Hay bạn hái trước (B). ');
Repeat
    Ch:=readkey;
Until Ucase(Ch) in ['M','B'];
Writeln;
If Ucase(Ch)='M' then Mayhai:=true
Else Mayhai:=false;
While Sobonghoa<>0 do
Begin
    If Mayhai then
    Begin
        If (Sobonghoa mod (Motlanhai+1))=0 then
            Dahai:=1
        Else Dahai:=(Sobonghoa mod (Motlanhai+1));
        Writeln(' - Máy hái ',Dahai,' bông hoa ');
    End
    Else
    Begin
        Repeat
            Write(' - Bạn hái bao nhiêu bông hoa ');
            Readln(Dahai);
        
```



```

        If (Dahai<=0)or(Dahai>Motlanhai)
        then Writeln(#7,'Bạn đã sai luật');
        Until (0<Dahai)and(Dahai<=Motlanhai);
    End;
    Mayhai:=not Mayhai;
    Dec(Sobonghoa,Dahai);
    Writeln('Còn lại ',Sobonghoa,' bông hoa');
End;
If Mayhai then Writeln('Bạn đã thắng ')
Else Writeln('Bạn đã thua. ');
Write('Tiếp tục chơi (C/K)?-');
Repeat
    tt:=readkey;
    tt:=Upcase(tt);
    Until tt in ['C','K'];
    If tt='C' then Clrscr;
End;
End;
(*=====*)
Begin { Main Program }
    Clrscr;
    Play;
End.

```

14. Program P4514;

```

Uses Crt;
Const M=30;
Var  a: array[1..M,1..M] of real;
      N, i, j, k: byte;

```

```
max1, max2, max3: Real;
(*=====*)
Procedure nhap(n: byte);
Begin
  For i:=1 to n do
    For j:=1 to n do
      Begin
        Write('a['',i','',j,']='); Readln(a[i,j]);
      End;
    End;
  End;
(*=====*)
Procedure Max_ab (Var m1,m2:real);
Begin
  k:=n;
  m1:=a[1,1];
  m2:=a[n,n];
  For i:=1 to n do
    Begin
      j:=1;
      Repeat
        If m1<a[i,j] then m1:=a[i,j];
        If m2<a[n-i+1,n-j+1] then m2:=a[n-i+1,n-j+1];
        inc(j);
      Until j>k;
      dec(k);
    End;
  End;
(*=====*)
```

```
Procedure Max_cde(Var mc, md, me: real);
Var h, bd, kt: byte;
Begin
  h:=(n+1) div 2;
  bd:=1;
  kt:=n;
  mc:=a[1,1];
  md:=a[n,n];
  me:=a[1,n];
  For i:=1 to h do
    Begin
      j:=bd;
      Repeat
        If mc<a[i,j] then mc:=a[i,j];
        If md<a[n-i+1,n-j+1] then
          md:=a[n-i+1,n-j+1];
        If me<a[j,n-i+1] then me:=a[j,n-i+1];
        inc(j);
      Until j>kt;
      inc(bd);
      dec(kt);
    End;
  End;
  (*=====*)
Begin
  Clrscr;
  Writeln('Nhập bảng số vuông N*N ');
  Write('N='); Readln(N);
```

```
nhap(N);  
Writeln('===Kết quả===');  
max_ab(max1,max2);  
Writeln('a. Max=', max1:0:2);  
Writeln('b.Max=', max2:0:2);  
max_cde(max1, max2, max3);  
Writeln('c.Max=', max1:0:2);  
Writeln('d.Max=', max2:0:2);  
Writeln('e.Max=', max3:0:2);  
Write('Nhấn Enter để kết thúc');  
Readln;
```

End.

15. Program P4515;

```
Uses Crt;  
Type Kieu=-1..1;  
Var N: longint;  
(*=====*)  
Procedure nhap(Var so: longint);  
Begin  
  Repeat  
    Write('N='); Readln(so);  
    If so<=0 then Write('nhập lại N>0 là số tự nhiên.');  Until so>0;  
End;  
(*=====*)  
Function nguyento(Ns: longint): boolean;  
{Trả về giá trị True nếu Ns là số nguyên tố, ngược lại trả  
về giá trị False }
```

```

Var kt: boolean;
    k: longint;
Begin
    If Ns in [2,3] then kt:=true
    Else
        Begin
            k:=2;
            kt:=true;
            While kt and (k<=Ns div 2) do
                Begin
                    If Ns mod k=0 then kt:=false
                    Else inc(k);
                End;
            End;
            nguyento:=kt;
        End;
    (*=====*)
Function Mobius(NS: longint): Kieu;
Var u, LN: longint;
    dem, dl: integer;
    kq: Kieu;
Begin
    LN:=NS;
    dem:=0;
    If NS=1 then kq:=1
    Else If NS in [2,3] then kq:=-1
    Else
        Begin

```

```

u:=2;
While (u<=(LN div 2))and(u<=NS) do
  Begin
    If (NS mod u=0)and nguyento(u) then
      {u là ước số nguyên tố của NS }
      Begin
        inc(dem);
        dl:=0;
        Repeat
          inc(dl);
          NS:= NS div u;
        Until (NS mod u<>0);
      End;
    inc(u);
  End; { While.... }
Case dem of
  0:kq:=-1;
  1:If dl>1 then kq:=0
    Else {Dem>=2 }
    If dem mod 2=0 then kq:=1
    Else kq:=-1;
End; { of Case...}
End;
Mobius:=kq;
End;
(*=====*)
Begin
  Clrscr;

```

```

WriteLn('Nhập số tự nhiên N');
nhap(N);
Write('M(',N,')=', Mobius(N));
ReadLn;
End.

```

16. a. Ta có:

$$C(2n, 2) = \frac{(2n)!}{2!(2n-2)!} = \frac{2n(2n-1)}{2} = n(2n-1) \\ = n(n-1) + n^2 = 2C(n, 2) + n^2$$

$$\begin{aligned} \text{b. } C(n, k) \cdot C(k, m) &= \frac{n!}{k!(n-k)!} \cdot \frac{k!}{m!(k-m)!} = \\ &= \frac{n!}{(n-k)! \cdot m! \cdot (k-m)!} = \frac{n!}{m!(n-m)!} \cdot \frac{(n-m)!}{(k-m)! \cdot (n-k)!} \\ &= C(n, m) \cdot C(n-m, k-m) \end{aligned}$$

$$\begin{aligned} \text{c. } C(m+n, 2) &= \frac{(m+n)(m+n-1)}{2} = \frac{m(m-1)}{2} + \frac{n(n-1)}{2} + mn \\ &= C(m, 2) + C(n, 2) + mn \\ &\Rightarrow C(m+n, 2) - C(m, 2) - C(n, 2) = mn. \end{aligned}$$

17. Từ đồng nhất thức: $(1+p)^{p+q} = (1+p)^p \cdot (1+p)^q$

Xét hệ số của x^r ở cả hai vế, ở vế trái hệ số này bằng $C(p+q, r)$, còn ở vế phải hệ số này bằng:

$$\sum_{j=0}^r C(p, j) C(q, r-j)$$

$$\text{Từ đó ta thu được } C(p+q, r) = \sum_{j=0}^r C(p, j) C(q, r-j)$$

b. Ta chứng minh công thức:

$$C(n + r + 1, r) = \sum_{j=0}^r C(n + j, j)$$

bằng qui nạp toán học theo r : Với $r=0,1$ công thức hiển nhiên đúng. Giả sử công thức đúng cho $r \geq 1$ ta chứng minh nó cũng đúng cho $r+1$, ta có:

$$\begin{aligned} \sum_{j=0}^{r+1} C(n + j, j) &= \sum_{j=0}^r C(n + j, j) + C(n + r + 1, r + 1) \\ &= C(n + r + 1, r) + C(n + r + 1, r + 1) \\ &= C(n + r + 2, r + 1) \end{aligned}$$

Điều phải chứng minh.

18. Program P4518;

Uses Crt;

Const Max=100;

Type khoang=0..Max;

Var a: array[1..Max] of khoang;

N, i, k, j, m, nl: khoang;

Dn: integer;

dk, stop: boolean;

(*=====*)

Function check: boolean;

{ check=TRUE nếu $a_1, a_2, \dots, a_n$ là 1 hoán vị không cân bằng, ngược lại check=FALSE }

Var kt: boolean;

id: khoang;

Begin


```
        kt:=true;
        id:=1;
        While kt and (id<=N) do
            If a[id]=id then kt:=false Else inc(id);
        check:=kt;
    End;
    (*=====*)
    Procedure doicho(Var so1, so2: khoang);
    {Đổi giá trị của hai biến so1 và so2 cho nhau}
    Begin
        so1:=so1+so2;
        so2:=so1-so2;
        so1:=so1-so2;
    End;
    (*=====*)
    Procedure print;
    { In ra một hoán vị không cân bằng}
    Var id: khoang;
    Begin
        For id:=1 to N do Write(a[id]:4);
        Writeln;
    End;
    (*=====*)
    Begin { Main Program }
        Clrscr;
        Write('Nhập N>=2 :');
        Repeat
            Readln(N);
```

```

    If  $N < 2$  then Write(#7, 'Nhập lại(  $N \geq 2$  ):');
Until  $N \geq 2$ ;
For  $i := 1$  to  $N$  do  $a[i] := i$ ;
stop := false; {Chưa đúng}
Repeat
     $i := N - 1$ ;
    While  $a[i] > a[i + 1]$  do
        If  $i > 1$  then dec(i);
     $j := N$ ;
    While  $a[i] > a[j]$  do If  $j > 1$  then dec(j);
    doicho(a[i], a[j]);
     $nl := N$ ;
     $m := i + 1$ ;
    While  $nl > m$  do
        Begin
            doicho(a[nl], a[m]);
            dec(nl);
            inc(m);
        End;
     $dk := true$ ;
     $i := 1$ ;
    Repeat
         $dk := (a[i] = a[i + 1] + 1)$ ; inc(i);
    Until ( $i = N$ ) or (not dk);
    If check = true then
        Begin
            inc(Dn); print;
        End;
    stop := (dk) and ( $i = N$ );
Until stop = true;

```

```

WriteLn('Có ',n,' hoán vị không cân bằng của ',N,
        ' phần tử');
Write('Nhấn Enter để thoát...');
ReadLn;
End.

```

19. Ta có $D_1=0$; $D_2=1$; $D_3=2$. Ta sẽ chứng minh:

$$D_n = (n-1)(D_{n-1} + D_{n-2}) \quad (1)$$

bằng qui nạp theo n .

Với $n=3$ thì (1) có dạng $2=(2-1)(1-0)$ - đúng.

Giả sử (1) đúng đến n , ta sẽ chứng minh (1) đúng đến $n+1$.

Ta xét 1 hoán vị không cân bằng $(a_1, a_2, \dots, a_n, a_{n+1})$ của $n+1$ phần tử. Rõ ràng $a_{n+1} \neq n+1$. Do đó tồn tại i ($1 \leq i \leq n$) để $a_i = n+1$. Có thể xảy ra hai khả năng:

- *Khả năng thứ nhất:* $a_{n+1} < i$. Khi đó nếu bỏ $a_i = n+1$ đi và thay vào đó bằng a_{n+1} thì ta được một hoán vị không cân bằng của n phần tử. Nói cách khác, hoán vị không cân bằng của $n+1$ phần tử có thể nhận được từ hoán vị không cân bằng của n phần tử bằng cách xếp $n+1$ vào vị trí thứ i và đổi a_i ra vị trí cuối cùng. Vì i có thể chọn từ 1 đến n nên số các hoán vị như vậy là nD_n .

- *Khả năng thứ hai:* $a_{n+1} = i$. Khi đó nếu bỏ cả a_i và a_{n+1} đi thì ta được một hoán vị không cân bằng của $(n-1)$ phần tử. Vì $1 \leq i \leq n$ nên ta có tất cả nD_{n-1} các hoán vị như vậy.

Vậy số các hoán vị không cân bằng của $n+1$ phần tử là:
 $D_{n+1} = nD_n + nD_{n-1} = n(D_n + D_{n-1})$ - điều phải chứng minh.

20. Ta chứng minh: $D_n = nD_{n-1} + (-1)^n$ (2) bằng qui nạp theo n .

Với $n=2$: (2) có dạng $1=1.0+(-1)^2$ - đúng.

Giả sử (2) đúng đến n , ta sẽ chứng minh (2) đúng đến $n+1$, nghĩa là $D_{n+1} = (n+1)D_n + (-1)^{n+1}$ (3).

Thực vậy:

$$\begin{aligned} \text{Vế trái của (3)} &= D_{n+1} = n(D_n + D_{n-1}) \quad (\text{theo (1) ở bài trên}) \\ &= nD_n + nD_{n-1} \\ &= nD_n + (D_n - (-1)^n) \quad (\text{theo giả thiết qui nạp}) \\ &= (n+1)D_n + (-1)^{n+1} = \text{vế phải} - \end{aligned}$$

Điều phải chứng minh.

§6. LÀM VIỆC VỚI FILE TRONG PASCAL

1. Program P4601;

Uses Crt;

Var f: text; {khai báo biến kiểu file văn bản}

m, n: integer;

Begin

Clrscr;

Assign(f, 'son.inf'); {hai thủ tục mở file}

Reset(f); {SON.INF để đọc}

Readln(f, m, n); {đọc một dòng vào hai biến m, n}

Close(f); {đóng file}

WriteLn(m, ' ', n);

Readln;

End.


```

close(f);
For i:=1 to n-1 do           {sắp xếp mảng}
  Begin
    jmax:=i;
    For j:=i to n do
      If (a[j]>a[jmax]) then j:=jmax;
    atg:=a[i];
    a[i]:=a[jmax];
    a[jmax]:=atg;
  End;
For i:=1 to n do WriteLn(a[i]);
ReadLn;
End.

```

4. Program P4604;

```

Uses Crt;
Var f: text;
    n, i: integer;
    a: array[1..100] of integer;
Begin
  Clrscr;
  assign(f,'data.txt');
  reset(f);
  ReadLn(f,n);
  For i:=1 to n do Read(f,a[i]);   {đọc các số trên
                                     cùng một dòng}

  Close(f);
  For i:=1 to n do Write(a[i]:8);
  ReadLn;
End.

```

5. Tự giải.

6. Program P4606;

Uses Crt;

Var f: text;

n, i: integer;

a: array[1..100] of integer;

Begin

Clrscr;

assign(f,'data.txt');

reset(f);

i:=1;

{hàm SeekEof báo cho ta biết đã hết file}

While not SeekEof(f) do

Begin

Readln(f,a[i]);

inc(i);

End;

n:=i-1;

For i:=1 to n do Writeln(a[i]);

Readln;

End.

7. Program P4607;

Uses Crt;

Var f: text;

n, i: integer;

a: array[1..100] of integer;

Begin

Clrscr;

```

assign(f,'data.txt');
reset(f);
i:=0;
{hàm SeekEoLn(f) báo cho ta biết đã hết dòng}
While not SeekEoLn(f) do
    Begin
        inc(i); Read(f,a[i]);
    End;
n:=i;
For i:=1 to n do Write(a[i]:8);
Readln;
End.

```

8. Program P4608;

```

Uses Crt;
Var f: text;
    n, m, i, j: integer;
    a: array[1..100,1..100] of integer;
Begin
    Clrscr;
    assign(f,'data.txt');
    reset(f);
    Readln(f,n,m);
    For i:=1 to n do
        Begin
            For j:=1 to m do Read(f,a[i,j]);
            Readln(f);
        End;
    close(f);

```



```
    For i:=1 to n do
        Begin
            For j:=1 to m do Write(a[i,j]:8);
        End;
    Readln;
End.
```

9. Program P4609;

```
{ $R+ }
Uses Crt;
Var f: text;
    fn: string; {Tên file}
    n, i: integer;
    ht: array[1..100] of string[20];
    t: array[1..100] of string[3];
    xl: array[1..100] of char;
Begin
    Clrscr;
    Write('Tên file cần đọc: '); Readln(fn);
    assign(f,fn);
    reset(f);
    Readln (f,n);
    For i:=1 to n do
        Readln (f, ht[i], t[i], xl[i]);
    close(f); {Kiểm tra}
    For i:=1 to n do Writeln(ht[i],t[i]:4,xl[i]:3);
    Readln;
End.
```

```
10. Program P4610;
    Uses Crt;
    Var f: text;
        fn: string; {Tên file}
        n, i: integer;
        ht: array[1..100] of string[20];
        t: array[1..100] of string[3];
        xl: array[1..100] of char;
    Begin
        Clrscr;
        Write("Tên file cần đọc: ");Readln(fn);
        assign(f,fn);
        reset(f);
        i:=1;
        While not(eof(f)) do
            Begin
                Readln(f,ht[i],t[i],xl[i]); inc(i);
            End;
        n:=i-1;
        close(f); {kiểm tra}
        For i:=1 to n do Writeln(ht[i],t[i]:4,xl[i]:3);
        Readln;
    End.

11.1. Program P46111;
    Uses Crt;
    Var f: text;
        m, n: integer;
```

```

Begin
  Clrscr;
  m:=10;
  n:=4;
  assign(f, 'son.inf');    {hai thủ tục mở file}
  rewrite(f);             {SON.INF để ghi}
  writeln(f, m, ', ', n);  {ghi giá trị các biểu thức m,
                           n vào một dòng của file}
  close(f);               {đóng file}
  assign(f, 'son.inf');    {đọc lại để kiểm tra}
  reset(f);
  readln(f, m, n);
  close(f);
  writeln(m:4, n:4);
  readln;
End.

```

11. 2. Program P46112;

```

Uses Crt;
Var f: text;
    m, n: integer;
Begin
  Clrscr;
  m:=4;
  n:=5;
  assign(f, 'sod.inf');
  rewrite(f);
  writeln(f, m);           {ghi m vào dòng đầu của file}
  writeln(f, n);           {ghi n vào dòng sau của file}
  close(f);

```

```
assign(f, 'sod.inf');  
reset(f);  
Readln(f, m);  
Readln(f, n);  
close(f);  
Writeln(m);  
Writeln(n);  
Readln;
```

End.

11. 3. Program P46113;

```
Uses Crt;  
Var f: text;  
    n, i: integer;  
    a: array[1..100] of integer;  
Begin  
    Clrscr;  
    Write('Nhập số n: ');Readln(n);  
    Writeln('Nhập dãy y n số : ');  
    For i:=1 to n do  
        Begin  
            Write('a[',i:2,'] = ');Readln(a[i]);  
        End;  
    assign(f, 'data.txt');  
    rewrite(f);  
    Writeln(f, n);    {ghi n vào dòng đầu của file}  
    For i:=1 to n do Writeln(f, a[i]);  
                    {ghi a[i] vào các dòng sau của file}  
    close(f);
```

```
assign(f, 'data.txt');  
reset(f);  
Readln(f, n);  
For i:=1 to n do Readln(f,a[i]);  
close(f);  
For i:=1 to n do Writeln(a[i]);  
Readln;  
End.
```

11. 4. Program P46114;

```
Uses Crt;  
Var f: text;  
    n, i: integer;  
    a: array[1..100] of integer;  
Begin  
    Clrscr;  
    Write('Nhập số n: ');Readln(n);  
    Writeln('Nhập dãy y n số: ');  
    For i:=1 to n do  
        Begin  
            Write('a[',i:2,'] = ');Readln(a[i]);  
        End;  
    assign(f,'data.txt');  
    rewrite(f);  
    Writeln(f,n);  
    For i:=1 to n do Write(f,a[i], ' '); {ghi các a[i] vào cùng  
                                         một dòng của file}  
    Writeln(f);  
    close(f);
```

```
assign(f,'data.txt');  
reset(f);  
Readln(f,n);  
For i:=1 to n do Read(f,a[i]);  
close(f);  
For i:=1 to n do Writeln(a[i]);  
Readln;  
End.
```

11. 6. Program P46116;

```
Uses Crt;  
Var f: text;  
    n, i: integer;  
    a: array[1..100] of integer;  
Begin  
    Clrscr;  
    Write('Nhập số n: '); Readln(n);  
    Writeln('Nhập một dãy y n số: ');  
    assign(f,'data.txt');  
    reWrite(f);  
    i:=1;  
    Repeat  
        Write('a',i:2,' = '); Readln(a[i]);  
        Writeln(f,a[i]);  
        inc(i);  
    Until (i>n);  
    close(f);  
    assign(f,'data.txt');
```

```
reset(f);  
While not(seekeoln(f)) do  
    Begin  
        Readln(f,a[i]); Writeln(a[i]);  
    End;  
close(f);  
Readln;  
End.
```

11. 9. Program P46119;

```
{ $R+ }  
Uses Crt;  
Var f: text;  
    fn: string; {Tên file}  
    n, i, l, j: integer;  
    ht: array[1..100] of string[20];  
    t: array[1..100] of string[3];  
    xl: array[1..100] of char;  
Begin  
    Clrscr;  
    Write('Tên file cần đọc: '); Readln(fn);  
    Write('Số lượng học sinh: '); Readln(n);  
    assign(f,fn);  
    rewrite(f);  
    Writeln(f,n);  
    For i:=1 to n do  
        Begin  
            Write('Nhập họ và tên: '); Readln(ht[i]);
```

```

        l:=length(ht[i]);
        If (l<20) then
            For j:=l+1 to 20 do ht[j]:=ht[i]+' ';
        Write('Nhập tuổi: '); Readln(t[i]);
        l:=length(t[i]);
        If (l<3) then
            For j:=l+1 to 3 do t[j]:=' '+t[i];
        Write('Nhập xếp loại: '); Readln(xl[i]);
        Writeln(f, ht[i], t[i], xl[i]);
    End;
close(f);
assign(f, fn);
reset(f);
Readln(f, n);
For i:=1 to n do
    Begin
        Readln(f, ht[i], t[i], xl[i]);
        Writeln(ht[i], t[i]:4, xl[i]:3);
    End;
close(f);
Readln;
End.

```

12. Được.

13. Program P4613;

Uses crt;

Var fin, fou: text;

n, m, i, j, k, kmax: integer;


```
    a: array[1..100] of real;
    atg: real;
Begin
    Clrscr;
    assign(fin, 'input.txt');
    reset(fin);
    assign(fou, 'output.txt');
    reWrite(fou);
    Readln(fin, n);
    Writeln(fou, n);
    For i:=1 to n do
        Begin
            j:=1;  {đọc dòng thứ i của input.txt}
            While not(seekeoln(fin)) do
                Begin
                    Read(fin, a[j]); inc(j);
                End;
            Readln(fin);
            {sắp xếp lại dòng thứ i đó}
            m:=j-1;
            For j:=1 to m-1 do
                Begin
                    kmax:=j;
                    For k:=j to m do
                        If {a[k]>a[kmax]} then kmax:=k;
                    atg:=a[j];
                    a[j]:=a[kmax];
                    a[kmax]:=atg;
```

```
End;
For j:=1 to m do {ghi dòng đó ra output.txt}
  Write(fou,a[j]:8:2);
  Writeln(fou);
End;
close(fin);
close(fou);
{kiểm tra}
assign(fou,'output.txt');
reset(fou);
Readln(fou,n);
For i:=1 to n do
  Begin
    While not(seekeofn(fou)) do
      Begin
        Read(fou,atg); Write(atg:8:2);
      End;
    Readln(fou); Writeln;
  End;
close(fou);
Readln;
End.
```

14. Program P4614;
Uses crt;
Var tfin, tfou: string;
 fin, fou: text;
 n, i, j: integer;

```
a: array[1..100] of string[23];
atg: string[23];
Begin
  Clrscr;
  Write('Đọc số liệu từ file:'); Readln(tfin);
  assign(fin, tfin);
  reset(fin);
  i:=1; {đọc file tfin}
  While not(seekeof(fin)) do
    Begin
      Readln(fin,a[i]); inc(i);
    End;
  {sắp xếp lại}
  n:=i-1;
  For i:=1 to n-1 do
    If (copy(a[i],21,3)='Nam') then
      For j:=i to n do
        If (copy(a[j],21,3)>='Nu') then
          Begin
            atg:=a[i]; a[i]:=a[j]; a[j]:=atg;
          End;
      close(fin);
      Write('Ghi số liệu ra file: '); Readln(tfou);
      assign(fou,tfou);
      reWrite(fou);
      For i:=1 to n do Writeln(fou, a[i]);
      close(fou);
      {kiểm tra}
```

```
    assign(fou,tfou);
    reset(fou);
    For i:=1 to n do
        Begin
            Readln(fou,atg); Writeln(atg);
        End;
    close(fou);
    Readln;
End.
```

15. Program P4615;

```
Uses Crt;
Var  tfin: string;
     fin: text;
     n, i, j, snam, snu: integer;
     a: array[1..100] of string[23];
     t: boolean;

(*=====*)
Function ho(hvt:string):string;
Var  i: byte;
     h: string;
Begin
    i:=1;
    While (hvt[i]=' ') do inc(i);
    h:="";
    While (hvt[i]<>' ') do
        Begin
            h:=h+upcase(hvt[i]); inc(i);
```

```

        End;
        ho:=h;
    End;
    (*=====*)
Begin
    Clrscr;
    Write('Đọc số liệu từ file: '); Readln(tfin);
    assign(fin, tfin);
    reset(fin);
    i:=1;      {đọc file tfin}
    While not(seekeof(fin)) do
        Begin
            Readln(fin, a[i]); inc(i);
        End;
    close(fin);
    {tính số nam, nữ}
    n:=i-1;
    snam:=0;
    snu:=0;
    For i:=1 to n do
        If (copy(a[i],21,3)='Nam') then snam:=snam+1
        Else snu:=snu+1;
    Writeln('Số nam: ',snam,'. Số nữ: ',snu,'. ');
    For i:=1 to n do {Tìm người chung họ}
        Begin
            j:=1;
            t:=false;
            Repeat

```

```

        If (ho(a[j])=ho(a[i]))and(j<>i) then t:=true;
        inc(j);
    Until (t)or(j>n);
    If t then Writeln(a[i]);
End;
Readln;
End.

```

16. Program P4616;

```

Uses Crt;
Var   tfin, tfou: string;
      fin, fou: text;
      n, i, j: integer;
      a: array[1..100] of string[23];
      atg: string[23];

(*=====*)
Function thutu(hvt:string): string;
Var   j, l, l1: byte;
      ten: string[8];
      g, tg: char;

Begin
    For i:=1 to 23 do hvt[i]:=upcase(hvt[i]);
    If (copy(hvt,21,3)='NAM') then g:='b'
    Else g:='a';
    ten:=' ';
    i:=20;
    While (hvt[i]=' ') do dec(i);
    j:=1;

```

```

While (hvt[i]<>' ') do
  Begin
    ten[j]:=hvt[i];
    dec(i);
    inc(j);
  End;
l1:=i;
l:=j-1;
For i:=1 to (l div 2) do
  Begin
    tg:=ten[i];
    ten[i]:=ten[l-i+1];
    ten[l-i+1]:=tg;
  End;
thutu:=g+ten+copy(hvt,1,l1);
End;
(*=====*)
Begin
  Clrscr;
  Write('Đọc số liệu ghi từ file: '); Readln(tfin);
  assign(fin, tfin);
  reset(fin);
  i:=1;      {Đọc file tfin}
  While not(seekeof(fin)) do
    Begin
      Readln(fin, a[i]); inc(i);
    End;
  n:=i-1;

```

```
For i:=1 to n-1 do
  For j:=i+1 to n do
    If (thutu(a[i])>thutu(a[j])) then {sắp xếp lại}
      Begin
        atg:=a[i];a[i]:=a[j];a[j]:=atg;
      End;
  close(fin);
  Write('Ghi số liệu ra file: '); Readln(tfou); {ghi ra file
  tfou}
  assign(fou, tfou);
  rewrite(fou);
  For i:=1 to n do WriteIn(fou, a[i]);
  close(fou);
  assign(fou,tfou); {kiểm tra}
  reset(fou);
  For i:=1 to n do
    Begin
      Readln(fou, atg); WriteIn(atg);
    End;
  close(fou);
  Readln;
End.
```

```
17. Program P4617;
Uses Crt;
Const st1='11';
      st2='010';
Var fin, fou: text;
    i, p: integer;
```



```
        st: string;
(*=====*)
Procedure ghi1(i:integer);
Begin
    st[i]:='0';
    st[i+1]:='0';
    Writeln(fou, st);
End;
(*=====*)
Procedure ghi2(i:integer);
Begin
    st[i+1]:='0';
    Writeln(fou, st);
End;
(*=====*)
Begin
    Clrscr;
    assign(fin, 'input.txt');
    reset(fin);
    Readln(fin, st);
    close(fin);
    l:=length(st);
    If ((st[l]='1')and(st[l-1]='0'))or((st[1]='1')and(st[2]='0')) then
        Writeln('Không thể biến đổi được!')
    Else
        Begin
            assign(fou, 'output.txt');
            rewrite(fou);
```

```

Writeln(fou,st);
If (st[l]='1')and(st[l-1]='1') then ghi1(l-1);
Repeat      { biến đổi xâu con '11' }
    p:=pos(st1,st);
    If (p<>0) then ghi1(p);
Until (p=0);
Repeat      { biến đổi xâu con '010' }
    p:=pos(st2,st);
    If (p<>0) then ghi2(p);
Until (p=0);
close(fou);
{kiểm tra}
assign(fou,'output.txt');
reset(fou);
While not(seekeof(fou)) do
    Begin
        Readln(fou,st); Writeln(st);
    End;
close(fou);
End;
Readln;
End.

```

18. Program P4618;

```

Uses Crt;
Var f: text;
    st: string;
    sb, se, sm, sd: string;

```

```
      b, e, m, d: integer
(*=====*)
Function dem(s1, s2: string): byte;
Var i, l, p, s: byte;
Begin
    l:=length(s2);
    For i:=1 to l do s2[i]:=upcase(st[i]);
    p:=pos(s1,s2);
    s:=0;
    While (p<>0) do
        Begin
            s:=s+1;
            l:=length(s2);
            s2:=copy(s2,p+1,l-p);
            p:=pos(s1,s2);
        End;
    dem:=s;
End;
(*=====*)
Begin
    Clrscr;
    assign(f,'bl.pas');
    reset(f);
    sb:='Begin';
    se:='End';
    sm:='{';
    sd:='}';
    b:=0;
```

```

e:=0;
m:=0;
d:=0;
While not(seekeof(f)) do
  Begin
    Readln(f,st);
    b:=b+dem(sb,st);
    e:=e+dem(se,st);
    m:=m+dem(sm,st);
    d:=d+dem(sd,st);
  End;
close(f);
Writeln(' b = ',b,' e = ',e,' m = ',m,' d = ',d);
If (b=e)then
  Writeln('Số các từ "Begin" bằng số các từ "End" .');
Else
  Writeln(' Số các từ "Begin" không bằng số các từ "End" .');
If (m=d)then
  Writeln('Số các dấu "{" bằng số các dấu "}" .');
Else
  Writeln('Số các dấu "{" không bằng số các dấu "}" .');
Readln;
End.

```

19. Program P4619;
 Uses Crt;
 Var fin, fou: text;
 st, st1: string;

```
l, p: byte;
Begin
  Clrscr;
  assign(fin, 'p4216.pas');
  reset(fin);
  assign(fou, 'blmoi.pas');
  rewrite(fou);
  st1:="";
  While not(seekeof(fin)) do
    Begin
      Readln(fin,st);
      st1:=st1+' '+st;
      Repeat
        l:=length(st1);
        p:=pos(',', st1);
        If (p<>0) then
          Begin
            st:=copy(st1, 1, p);
            Writeln(fou, st);
            st1:=copy(st1, p+1, l-p);
          End;
      Until (p=0);
    End;
  Writeln(fou, st1);
  close(fin);
  close(fou);
  assign(fou, 'blmoi.pas');
  reset(fou);
```

```
While not(seekeof(fou)) do
  Begin
    Readln(fou, st); Writeln(st);
  End;
close(fou);
Readln;
End.
```

20. Program P4620;

```
Uses Crt;
Var fin, fou, fhuy: text;
    sum, can: real;
    n, i, k: byte;
Begin
  Clrscr;
  assign(fin, 'KHACH.INF');
  reset(fin);
  Readln(fin, n);
  assign(fou, 'CAN.KQ');
  rewrite(fou);
  assign(fhuy, 'HUY.KQ');
  rewrite(fhuy);
  For i:=1 to n do
    Begin
      sum:=0;
      k:=0;
      While not(seekeoln(fin)) do
```

```
Begin
    Read(fin,can);
    sum:=sum+can;
    k:=k+1;
End;
Writeln(fou, sum);
If (sum>20)or(k>10) then Writeln(fhuy, i);
Readln(fin);
End;
close(fin);
close(fou);
close(fhuy);
assign(fou, 'CAN.KQ');
reset(fou);
Writeln('File "CAN.KQ" ');
While not(seekeof(fou)) do
    Begin
        Readln(fou, sum); Writeln(sum:6:2);
    End;
close(fou);
assign(fhuy, 'HUY.KQ');
reset(fhuy);
Writeln('File "HUY.KQ" ');
While not(seekeof(fhuy)) do
    Begin
        Readln(fhuy,i); Writeln(i);
    End;
close(fhuy);
Readln;
End.
```

§7. ÔN LUYỆN

1. Program P4701;

Uses Crt;

Const Pi: real = 3.14;

Var r, a, h1, h2: real;

chon: integer;

(*=====*)

Procedure dien_tich_hinh_tron (r: real);

Var dt: real;

Begin

dt:=Pi*r*r;

WriteLn('Diện tích hình tròn là:', dt);

End;

(*=====*)

Procedure dien_tich_hinh_vuong (a: real);

Var dt: real;

Begin

dt:=a*a;

WriteLn('Diện tích hình vuông là:', dt);

End;

(*=====*)

Procedure dien_tich_hinh_thoi (h1, h2: real);

Var dt:=real;

Begin

dt:=h1*h2/2;

WriteLn('Diện tích hình thoi là:', dt);

End;


```

(*=====*)
Begin    {Main Program}
    Clrscr;
    Writeln('Nhập bán kính hình tròn:'); Readln(r);
    Writeln('Nhập cạnh hình vuông:'); Readln(a);
    Writeln('Nhập đường chéo hình thoi: h1, h2');
    Readln(h1, h2);
    Writeln('Bạn muốn tính diện tích hình gì:');
    Writeln('1 - Tính diện tích hình tròn. ');
    Writeln('2 - Tính diện tích hình vuông. ');
    Writeln('3 - Tính diện tích hình thoi. ');
    Writeln('Chọn 1, 2 hoặc 3: '); Readln(chon);
    Case chon of
        1: dien_tich_hinh_tron(r);
        2: dien_tich_hinh_vuong(a);
        3: dien_tich_hinh_thoi(h1, h2);
    End;
    Readln;
End.

```

2. Program P4702;

```

Uses Crt;
Var a, b, c: real;
(*=====*)
Procedure Giai_PT_bac_hai (a, b, c: real);
Var Delta, x1, x2: real;
(*=====*)
Procedure Delta_duong;
Begin

```

```

x1:=(-b+sqrt(delta))/(2*a);
x2:=(-b-sqrt(delta))/(2*a);
Writeln('Phương trình có hai nghiệm đơn:');
Writeln('X1=', x1, ' X2=', x2);
End;
(*=====*)
Procedure Delta_khong;
Begin
    Writeln('Một nghiệm kép:');
    Writeln('X=', -b/(2*a));
End;
(*=====*)
Begin    {thân thủ tục chính}
    Delta:=b*b -4*a*c;
    If delta=0 then Delta_khong;
    If delta>0 then Delta_duong;
    If delta<0 then Write ('Phương trình vô nghiệm');
End.
(*=====*)
Begin    {chương trình chính}
    Clrscr;
    Write('Hệ số a='); Readln(a);
    Write('Hệ số b='); Readln(b);
    Write('Hệ số c='); Readln(c);
    Giai_pt_bac_hai (a, b, c);
End.

```

3. Program P4703;
Uses Crt;

```
Var a, b: integer;
    sum: word;
    A: array[1..16] of integer;
    i, j: integer;
Begin
    Clrscr;
    Write('Cho hai số tự nhiên a, b:') Readln(a, b);
    Sum:= a+b;
    i:=0;
    While sum>0 do
        Begin
            i:= i+1;
            A[i]:=Sum mod 2;
            Sum:=Sum div 2;
        End;
    WriteLn('Tổng của hai số tự nhiên a, b trong hệ '
            +'cơ số 2 là:');
    For j:=i downto 1 do Write(A[j]);
    Readln;
End.
```

4. Program P4704;

Uses Crt;

Var n: integer;

(*=====*)

Function ngto (n: integer): boolean;

Var i: integer;

Begin

i:=2;

```

    While (i<=sqrt(m)) and (n Mod i >0) do inc(i);
    If i>sqrt(n) then ngto:=true
    Else ngto:=False;
End;
(*=====*)
Function dx(n: integer): boolean;
Var A: array[1..16] of integer;
    i, j: integer;
Begin
    i:=0;
    While n>0 do
        Begin
            inc(i);
            A[i]:=n mod 2;
            n:=n div 2;
        End;
    j:=1;
    While (j<=i div 2) and (A[j]=A[i-j+1]) do inc(j);
    If j=(i div 2) + 1 then dx:=true
    Else dx:=False;
End.
(*=====*)
Begin {chương trình chính}
    Writeln('Các số tự nhiên thoả mã n là:');
    For n:=100 to 999 do
        Begin
            if (dx(n)) and (ngto(n)) then Writeln(n:4);
        End;
    Readln;
End.

```

```
5. Program P4705;
   Uses Crt;
   Var S, Word: String;
       L, i: integer;
   Begin
       Clrscr;
       Writeln('Cho một xâu kí tự:'); Readln(S);
       i:=length(S);
       Word:="";
       While (i>=1) and (S[i]<>#32) do
       Begin
           word:= S[i] + word;
           dec(i);
       End;
       Write('Từ cuối cùng của xâu là:'); Writeln(Word);
       j:=1;
       L:=Length(Word);
       While (j<=L div 2) and (Word[j]=Word[L-j+1]) do
           inc(j);
       If j>(L div 2) then Writeln('Đây là từ đối xứng')
       Else Writeln('Từ không đối xứng');
       Readln;
   End.
```

6. Để ý rằng nếu $2^n = 3^m + 1$ và $n \leq 10$ thì ta phải có
 $3^m = 2^n - 1 < 2^{10} = 1024 < 3^7$ suy ra $m \leq 6$.

```
Program P4706;
Uses Crt;
```

```

Var m, n: integer;
(*=====*)
Function Hai_mu (n: integer): integer;
  Var i, M: integer;
  Begin
    M:=1;
    For i:=1 to n do M:=M*2;
    Hai_mu:=M;
  End;
(*=====*)
Function Ba_mu (n: integer): integer;
  Var i, M: integer;
  Begin
    M:=1;
    For i:=1 to n do M:=M*3;
    Ba_mu:=M;
  End;
(*=====*)
Begin    {chương trình chính}
Clrscr;
  For n:=1 to 10 do
    For m:=1 to 6 do
      Begin
        If hai_mu(n)=ba_mu(m)+1 then Writeln(n:2, m:2);
      End;
    End;
  Readln;
End.

```

7. Program P4707;

Uses Crt;

Var A: String [10];

B: array [1..16] of integer;

gt: word;

l, i: integer;

Begin

Clrscr;

Write ('Cho một số trong hệ cơ số 9:'); Readln (A);

l:=length(A);

For i:=1 to l do

Begin

Gt:= ord (A[i]) - 48;

B[2*i-1]:= gt Mod 3;

B[2*i]:= gt Div 3;

End;

Writeln(Biểu diễn của số trong hệ cơ số 3 là:);

For i:=2*l downto 1 do Write (B[i]);

Readln;

End.

8. Program P4708;

Uses Crt;

Const so: array[0.. 15] of

char=('0','1','2','3','4','5','6','7',

'8','9','A','B','C','D','E','F');

Var N: longint;

A: array [1..16] of char;

i, j: byte;

```

Begin
  Clrscr;
  Write('Cho một số tự nhiên n:'); Readln (n);
  i:=0;
  While n>0 do
    Begin
      Inc(i);
      A[i]:= so[n mod 16];
      N:= n Div 16;
    End;
  Writeln ('biểu diễn của n trong hệ Hexa là: ');
  For j:=i downto 1 do Write(A[j]);
  Readln
End.

```

9. Program P4709;

```

Var M: Hexan: string[8];
    N, GTM: longint;
    F: text;
    i, j: integer;
    A: array [1..10] of integer;
(*=====*)
Procedure CHUYEN_HE_10;
Begin
  For i:= 1 to Length (M) do
    If (M[i]>='0') and (M[i]<='9') then A[i]:=ord(M[i])-48
    Else A [i]:= ord (M[i]) - 55;
  GTM:=0,

```



```

    For i:=1 to Length (M) do GTM:=16*GTM + A[i];
End;
(*=====*)
Procedure CHUYEN_HE_16;
Begin
    i:=0;
    While N>0 do
        Begin
            Inc(i);
            A[i]:=N mod 16;
            N:=N div 16;
        End;
    For J:=1 to i do
        Begin
            If A[i-j+1]<10 then Hexan [j]: = Char(A[i-j+1] + 48)
            Else Hexan[j]: = Char (A[i-j+1]+55);
        End;
    End;
(*=====*)
Begin {Chương trình chính}
    Write ('cho một số tự nhiên N:'); Readln (N);
    Wriet('cho một số tự nhiên M trong hệ đếm Hexa:');
    Readln(M);
    CHUYEN_HE_16;
    CHUYEN_HE_10;
    Assign (F, 'KQ12.txt'); ReWrite (F);
    Writeln (F,N);
    Writeln (F,M);

```

```

        Writeln (F,Hexan);
        Writeln (F, GTM);
        Close(F);
    End.

```

10. Program P4710;

```

    Uses Crt;
    Const Nmax=100;
    Type He_so=array[0..Nmax] of integer;
    Var Sn,n,i:integer;
        ng,a:He_so;
    (*=====*)
    Function F(x: integer): integer;
    {tính giá trị hàm f(x)=vế trái tại điểm x}
    Var id, S: integer;
    Begin
        S:=0;
        For id:=N downto 0 do S:=S*x+a[id];
        F:=S;
    End;
    (*=====*)
    Procedure Gpt;
    Begin
        sn:=0;
        For i:=-b[0] to b[0] do
            Begin
                If (i=0) then
                    Begin
                        If a[0]=0 then

```

```
        Begin inc(sn); ng[sn]:=0; End;
    End
Else { i<>0 }
    If (a[0] mod i=0)and(F(i)=0) then
        Begin
            inc(sn); ng[sn]:=i;
        End;
    End;
End;
if sn<>0 then
Begin
    Writeln('Phương trình có ', sn, ' nghiệm:');
    For i:=1 to sb do Writeln('X', i, '=', ng[i]);
End
Else Writeln('Phương trình vô nghiệm');
Write('Nhấn Enter để thoát...');
End;
(*=====*)
Begin
    Clrscr;
    Writeln('Giải phương trình bậc N');
    Write('N='); Readln(n);
    For i:=N downto 0 do
        Begin
            Write('a',i,'='); Readln(a[i]);
        End;
    Gpt;
    Readln;
End.
```

```

11. Program P4711;
    Uses Crt;
    Type np=array [0..79] of byte;
    Var a, b, c: np;
        n, i, e, f: byte;
    (*=====*)
    Procedure nhap(tenx: string;Var x: np);
    Var n, i, x0, code: integer;
        ch: char;
    Begin
        Writeln('Nhập số nhị phân',tenx,' : ');
        n:=-1;
        Repeat
            Repeat
                ch:=readkey;
            Until (ch in ['0','1','#13]);
            If not(ch=#13) then
                Begin
                    Write(ch);
                    inc(n);
                    If (n>0) then
                        For i:=n downto 1 do x[i]:=x[i-1];
                        val(ch,x0,code);
                        x[0]:=x0;
                End;
            Until (ch=#13);
            For i:=79 downto 0 do Write(x[i]);
        End;
    
```

```

(*=====*)
Begin
  Clrscr;
  nhap('a', a);
  nhap('b', b);
  e:=0;
  For i:=0 to 79 do
    Begin
      f:=(a[i]+b[i]+e) div 2;
      c[i]:=a[i]+b[i]+e-2*f;
      e:=f;
    End;
  For i:=79 downto 0 do Write(c[i]);
  Readln;
End.

```

12. Program P4712;

Uses Crt;

CONST maxnp=79;

Type np= array [0..maxnp] of byte;

Var a, b, c, p: np;

i, j: integer;

```

(*=====*)

```

Procedure nhap(tenx:string;Var x:np);

Var n, i, x0, code: integer;

ch: char;

Begin

Write(' Nhập số nhị phân', tenx, ':');

n:=-1;

```

Repeat
  Repeat
    ch:=readkey;
  Until (ch in ['0','1','#13]);
  If not(ch=#13) then
    Begin
      Write(ch);
      inc(n);
      If (n>0) then
        For i:=n downto 1 do x[i]:=x[i-1];
      val(ch,x0,code);
      x[0]:=x0;
    End;
  Until (ch=#13);
  Writeln;
End;

(*=====*)
Procedure left(x: np; k: byte; Var y: np);
Var i: integer;
Begin
  For i:=maxnp downto 0 do
    If (i>=k) then y[i]:=x[i-k]
    Else y[i]:=0;
  End;
  (*=====*)
Procedure cong(x,y: np; Var z: np);
Var i, e, f: integer;
Begin
  e:=0;

```

```

For i:=0 to maxnp do
  Begin
    f:=(x[i]+y[i]+e) div 2;
    z[i]:=x[i]+y[i]+e-2*f;
    e:=f;
  End;
End;
(*=====*)
Begin
  Clrscr;
  nhap('a', a);
  nhap('b', b);
  For i:=0 to maxnp do p[i]:=0;
  For i:=0 to maxnp do
    Begin
      If b[i]=1 then left(a,i,c)
      Else For j:=0 to maxnp do c[j]:=0;
      cong(p,c,p);
    End;
  Write('a*b=');
  For i:=maxnp downto 0 do Write(p[i]);
  Readln;
End.

```

13. Program P4713:

Uses Crt;

Type np=array [0..1000] of byte;

Var a, b, c, a0, a1, b0, b1, p, p0, p1: np;

```

maxnp, i, j, na, nb, n: integer;
(*=====*)
Procedure nhap(tenx: string; var x: np; var m: integer);
Var  n0, i, x0, dem, code: integer;
     ch: char;
Begin
  Write('Nhập số nhị phân ', tenx, ' : ');
  x[0]:=1;
  n0:=0;
  Repeat
    Repeat
      ch:=readkey;
    Until (ch in ['0', '1', #13]);
    If (ch in ['0', '1']) then
      Begin
        Write(ch);
        inc(n0);
        If (n0>1) then
          For i:=n0 downto 2 do x[i]:=x[i-1];
          val(ch, x0, code);
          x[1]:=x0;
        End;
      Until (n>=maxnp) or (ch=#13);
      m:=n0;
      Writeln;
    End;
  (*=====*)
Procedure left(x: np; k: byte; var y: np);

```



```
Var i: integer;
Begin
  For i:=maxnp downto 1 do
    If (i>k) then y[i]:=x[i-k] Else y[i]:=0;
End;
(*=====*)
Procedure cong(x, y: np; var z: np); {x,y>=0}
Var i, nho: integer;
Begin
  nho:=0;
  For i:=1 to maxnp do
    Begin
      z[i]:=(x[i]+y[i]+nho) mod 2;
      If (x[i]+y[i]+nho>=2) then nho:=1
      Else nho:=0;
    End;
  End;
(*=====*)
Procedure tru(x, y: np; var z: np); {x>=y>=0}
Var i, nho: integer;
Begin
  nho:=0;
  For i:=1 to maxnp do
    Begin
      z[i]:=(2+x[i]-y[i]-nho) mod 2;
      If (x[i]-y[i]-nho>=0) then nho:=0
      Else nho:=1;
    End;
```

```

End;
(*=====*)
Function daucong(var x, y: np): byte;
{xét dấu của tổng x+y sắp xếp theo trị tuyệt đối}
Var u, i, d: integer;
Begin
    i:=maxnp;
    While (x[i]=y[i]) and (i>0) do dec(i);
    If (i>=1) and (x[i]<y[i]) then
        Begin
            daucong:=y[0];
            For i:=0 to maxnp do
                Begin
                    u:=x[i];    x[i]:=y[i]; y[i]:=u;
                End;
            End
        End
    Else daucong:=x[0];
End;
(*=====*)
Procedure cong1(x, y: np; var z: np); { x,y nguyên }
Begin
    If (x[0]=y[0]) then
        Begin
            z[0]:=x[0]; cong(x,y,z);
        End
    Else
        Begin
            z[0]:=daucong(x, y); tru(x, y, z);

```

```
End;

End;

(*=====*)
Procedure tru1(x, y: np; var z: np); {x,y nguyên}
Begin
    If (y[0]=0) then y[0]:=1
    Else y[0]:=0;
    cong1(x, y, z);
End;

(*=====*)
Procedure nhan(x, y: np; var z: np); {x,y nguyên}
Var i, j: integer; u: np;
Begin
    If (x[0]=y[0]) then z[0]:=1 else z[0]:=0;
    For i:=1 to maxnp do z[i]:=0;
    For i:=1 to maxnp do
        Begin
            If y[i]=1 then left(x,i-1,u)
            Else
                For j:=1 to maxnp do u[j]:=0;
            cong(z,u,z);
        End;
    End;
End;

(*=====*)
Begin {Chương trình chính}
    Clrscr;
    maxnp:=79;
    nhap('a', a, na);
```

```

nhap('b', b, nb);
n:=nb;
If (na>n) then n:=na;
n:=(n+1) div 2;
For i:=1 to 2*n do
  Begin
    If (i<=n) then
      Begin
        a0[i]:=a[i]; b0[i]:=b[i];
      End
    Else
      Begin
        a1[i-n]:=a[i]; b1[i-n]:=b[i];
      End
    End;
  maxnp:=2*n;
  nhan(a0, b0, p0);
  tru1(a1, a0, a0);
  tru1(b0, b1, b0);
  nhan(a0, b0, a0);
  nhan(a1, b1, a1);
  maxnp:=2*maxnp;
  left(p0, n, p1);
  cong(p1, p0, p0);
  left(a0, n, p1);
  cong1(p0, p1, p0);
  left(a1, n, b1);
  cong(b1, a1, a1);

```

```

left(a1, n, b1);
cong(b1, p0, p);
Write('a*b = ');
While p[maxnp]=0 do dec(maxnp);
For i:=maxnp downto 1 do Write(p[i]);
Readln;
End.

```

14. Ta chứng minh $F_k F_n + F_{k+1} F_{n+1} = F_{n+k+1}$ (1) đúng với mọi k, n bằng quy nạp theo k :

Với $k=1$: (1) có dạng $1 \cdot F_n + 1 \cdot F_{n+1} = F_{n+2}$ - đúng với mọi n .

Giả sử (1) đúng đến k , nghĩa là $F_k F_n + F_{k+1} F_{n+1} = F_{n+k+1}$ với mọi n , mọi $k \leq k$; ta phải chứng minh (1) đúng đến $k+1$, nghĩa là:

$$F_{k+1} F_n + F_{k+2} F_{n+1} = F_{n+k+2} \quad (2).$$

$$\begin{aligned}
 \text{Thực vậy, vế trái của (2)} &= (F_k + F_{k-1}) F_n + (F_{k+1} + F_k) F_{n+1} \\
 &= F_k F_n + F_{k-1} F_n + F_{k+1} F_{n+1} + F_k F_{n+1} \\
 &= (F_k F_n + F_{k+1} F_{n+1}) + (F_{k-1} F_n + F_k F_{n+1}) \\
 &= F_{n+k+1} + F_{n+k} \quad (\text{theo giả thiết quy nạp}) \\
 &= F_{n+k+2} = \text{vế phải của (2)}.
 \end{aligned}$$

Vậy (1) đúng đến $k+1$, nghĩa là (1) đúng với mọi k , mọi n .

15. a. Ta chứng minh $F_1^2 + F_2^2 + \dots + F_n^2 = F_n F_{n+1}$ (1) bằng quy nạp theo n :

Với $n=1$: (1) có dạng $1^2 = 1 \cdot 1$ - đúng.

Giả sử (1) đúng đến n . Ta sẽ chứng minh (1) đúng đến $n+1$, nghĩa là:

$$F_1^2 + F_2^2 + \dots + F_n^2 + F_{n+1}^2 = F_{n+1}F_{n+2} \quad (2).$$

Thực vậy:

$$\begin{aligned} \text{Vế trái của (2)} &= (F_1^2 + F_2^2 + \dots + F_n^2) + F_{n+1}^2 \\ &= F_n F_{n+1} + F_{n+1}^2 \quad (\text{theo giả thiết quy nạp}) \\ &= F_{n+1}(F_n + F_{n+1}) \\ &= F_{n+1}F_{n+2} = \text{vế phải của (2)}. \end{aligned}$$

Vậy (1) đúng đến $(n+1)$ nghĩa là (1) đúng với mọi n .

b. Ta sẽ chứng minh $F_1 + F_3 + \dots + F_{2n-1} = F_{2n}$ (1) bằng quy nạp theo n :

Với $n=1$ thì (1) có dạng $1 = 1$ - đúng.

Giả sử (1) đúng đến n , nghĩa là $F_1 + F_3 + \dots + F_{2n-1} = F_{2n}$.

Ta sẽ chứng minh (1) đúng đến $(n+1)$, nghĩa là:

$$F_1 + F_3 + \dots + F_{2n-1} + F_{2(n+1)-1} = F_{2(n+1)} \quad (2).$$

$$\begin{aligned} \text{Thực vậy: vế trái của (2)} &= (F_1 + F_3 + \dots + F_{2n-1}) + F_{2n+1} \\ &= F_{2n} + F_{2n+1} \quad (\text{theo giả thiết quy nạp}) \\ &= F_{2n+2} = \text{vế phải của (2)} \end{aligned}$$

Vậy (1) đúng đến $(n+1)$. Do đó (1) đúng với mọi n .

16. a. Ta sẽ chứng minh $F_n + F_{n+2} = L_{n+1}$ (1) bằng quy nạp theo n :

Với $n=1$ thì (1) có dạng $1+2=3$ - đúng.

Giả sử (1) đúng đến n , nghĩa là $F_k + F_{k+2} = L_{k+1}$ với mọi $k \leq n$. Ta phải chứng minh (1) đúng đến $(n+1)$, nghĩa là:

$$F_{n+1} + F_{(n+1)+2} = L_{(n+1)+1} \quad (2).$$

Thực vậy: vế trái của (2) $= F_{n+1} + F_{n+3}$

$$= F_n + F_{n-1} + F_{n+2} + F_{n+1}$$

$$= (F_n + F_{n+2}) + (F_{n-1} + F_{n+1})$$

$$= L_{n+1} + L_n \text{ (theo giả thiết quy nạp)}$$

$$= L_{n+2} = \text{vế phải của (2)}.$$

Vậy (1) đúng đến $n+1$ suy ra (1) đúng với mọi n .

b. Ta sẽ chứng minh $L_0^2 + L_1^2 + L_2^2 + \dots + L_n^2 = L_n L_{n+1} + 2$ (1)
bằng quy nạp theo n :

Với $n=1$ thì (1) có dạng $2^2 + 1^2 = 1.3 + 2$ - đúng.

Giả sử (1) đúng đến n , ta phải chứng minh (1) đúng đến $n+1$, nghĩa là:

$$L_0^2 + L_1^2 + L_2^2 + \dots + L_n^2 + L_{n+1}^2 = L_{n+1} L_{n+2} + 2 \quad (2).$$

$$\text{Thực vậy: vế trái của (2)} = (L_0^2 + L_1^2 + L_2^2 + \dots + L_n^2) + L_{n+1}^2$$

$$= (L_{n+1} L_n + 2) + L_{n+1}^2 \text{ (theo giả thiết quy nạp)}$$

$$= L_{n+1} (L_n + L_{n+1}) + 2$$

$$= L_{n+1} L_{n+2} + 2 = \text{vế phải của (2)}.$$

Vậy (1) đúng đến $(n+1)$, suy ra (1) đúng với mọi n - điều phải chứng minh.

17. a. Ta chứng minh $F_{n+1} F_{n-1} - F_n^2 = (-1)^n$ (1) bằng quy nạp theo n :

Với $n=2$ thì (1) có dạng $2.1 - 1^2 = (-1)^2$ - đúng.

Giả sử (1) đúng đến (n). Ta chứng minh (1) đúng đến (n+1), nghĩa là:

$$F_{n+2}F_n - F_{n+1}^2 = (-1)^{n+1} \quad (2).$$

$$\begin{aligned} \text{Thực vậy: vế trái của (2)} &= (F_{n+1} + F_n)(F_{n+1} - F_{n-1}) - F_{n+1}^2 \\ &= F_{n+1}^2 + F_n F_{n+1} - F_n F_{n-1} - F_{n+1} F_{n-1} - F_{n+1}^2 \\ &= F_n(F_n + F_{n-1}) - F_n F_{n-1} - F_{n+1} F_{n-1} \\ &= F_n^2 + F_n F_{n-1} - F_n F_{n-1} - F_{n+1} F_{n-1} \\ &= (F_n^2 - F_{n+1} F_{n-1}) = -(-1)^n \quad (\text{theo giả thiết quy nạp}) \\ &= (-1)^{n+1} = \text{vế phải của (2)} \end{aligned}$$

Như vậy (1) đúng đến (n+1), do đó (1) đúng với mọi n.

b. Ta chứng minh $F_0 F_1 + F_1 F_2 + \dots + F_{2n-1} F_{2n} = F_{2n}^2$ (1) bằng quy nạp theo n:

Với n=1 thì (1) có dạng: $F_0 F_1 + F_1 F_2 = F_2^2$ hay $0+1.1 = 1^2$ - đúng.

Giả sử (1) đúng đến n, ta sẽ chứng minh (1) đúng đến (n+1), nghĩa là:

$$F_0 F_1 + F_1 F_2 + \dots + F_{2n-1} F_{2n} + F_{2n} F_{2n+1} + F_{2n+1} F_{2n+2} = F_{2n+2}^2 \quad (2).$$

$$\begin{aligned} \text{Thực vậy: vế trái của (2)} &= F_{2n}^2 + F_{2n} F_{2n+1} + F_{2n+1} F_{2n+2} \\ &\quad (\text{theo gt quy nạp}) \end{aligned}$$

$$= F_{2n}(F_{2n} + F_{2n+1}) + F_{2n+1} F_{2n+2}$$

$$= F_{2n} F_{2n+2} + F_{2n+1} F_{2n+2}$$

$$= F_{2n+2}(F_{2n} + F_{2n+1})$$

$$= F_{2n+2}^2 = \text{vế phải của (2)}.$$

Vậy (1) đúng đến $(n+1)$ nên (1) đúng với mọi n - điều phải chứng minh.

c. Ta sẽ chứng minh $F_0 - F_1 + F_2 - F_3 + \dots - F_{2n-1} + F_{2n} = F_{2n-1} - 1$
(1) bằng quy nạp theo n :

Với $n=1$ thì (1) có dạng $-1+1=1-1$ - đúng.

Giả sử (1) đúng đến (n) , ta sẽ chứng minh (1) đúng đến $(n+1)$. Thực vậy thay n bằng $(n+1)$ thì (1) sẽ thành:

$$F_0 - F_1 + F_2 - F_3 + \dots - F_{2n-1} + F_{2n} - F_{2n+1} + F_{2n+2} = F_{2n+1} - 1 \quad (2).$$

$$\text{Vế trái của (2)} = F_{2n-1} - 1 - F_{2n+1} + F_{2n+2} \quad (\text{theo gt quy nạp})$$

$$= F_{2n-1} - 1 - (F_{2n} + F_{2n-1}) + (F_{2n+1} + F_{2n})$$

$$= F_{2n+1} - 1 = \text{vế phải của (2)}$$

Vậy (1) đúng đến $(n+1)$ nên (1) đúng với mọi n - Điều phải chứng minh.

18. Program P4718b;

Uses Crt;

Var a: array [1..100] of integer;

i, j, k, l, dn: integer;

ok: boolean;

Begin

Clrscr;

Writeln('Tính 100 số hạng của dãy y P4717 :');

Writeln('Chờ 1 phút...');

a[1]:=1;

a[2]:=2;

i:=3;

Repeat

```
j:=a[i-1];
Repeat
    inc(j);
    ok:=false;
    dn:=0;
    For k:=1 to i-2 do
        Begin
            For l:=k+1 to i-1 do
                If (a[k]+a[l]=j) then
                    Begin
                        ok:=true; dn:=dn+1;
                    End;
            End;
        End;
    Until (ok)and(dn=1);
    a[i]:=j;
    inc(i);
Until (i>100);
For i:=1 to 100 do Write(a[i]:8);
Readln;
End.
```

19. Program P4719;

Uses Crt;

Type kthang=1..12;

Const

a: array [0..6] of string=('chunhat','hai','ba','tu','nam','sau','bay');

Var th, t1, t2: kthang;

nam, s, s1, n1, n2, nam1, nam2: integer;

i: longint;

```
(*=====*)
Function nhuan(nam: integer): boolean;
Var kt: boolean;
Begin
    If (nam mod 4=0) then
        If (nam mod 100=0) then
            If (nam mod 400=0) then kt:=true
            Else kt:=false
        Else kt:=true
    Else kt:=false;
    nhuan:=kt;
End;
(*=====*)
Function ngaynam(nam: integer): integer;
Begin
    If nhuan(nam) then ngaynam:=366
    Else ngaynam:=365;
End;
(*=====*)
Function ngaythang(thang: kthang;
                    nam: integer): integer;
Begin
    Case thang of
        4, 6, 9, 11: ngaythang:=30;
        1, 3, 5, 7, 8, 10, 12: ngaythang:=31;
        2: If nhuan(nam) then ngaythang:=29
           Else ngaythang:=28;
    End;
End;
```

```
(*=====*)
Function ngaytd(n1: integer; thang1: kthang; nam1:
                integer):longint;
Var n: longint;
    i: integer;
Begin
    n:=0;
    If (nam1>1) then
        For i:=1 to nam1-1 do n:=n+ngaynam(i);
    If (thang1>1) then
        For i:=1 to thang1-1 do n:=n+ngaythang(i,nam1);
    n:=n+n1;
    ngaytd:=n;
End;
(*=====*)
Begin
    Clrscr;
    Write('Nhập năm: ');Readln(nam);
    s:=0;
    For i:=ngaytd(1,1,nam) to ngaytd(31,12,nam) do
        If (i mod 7=0) then s:=s+1;
    Writeln(' Năm ',nam,' có ',s,' ngày chủ nhật');
    Writeln('Nhấn Enter tiếp tục ...');
    Writeln('Nhập hai ngày: ');
    Write('Ngày đầu (ngày tháng năm) : ');
    Readln(n1,t1,nam1);
    Write('Ngày cuối (ngày tháng năm) : ');
    Readln(n2,t2,nam2);
```

```
Writeln('Từ ngày ',n1,'/',t1,'/',nam1);
Writeln(' đến ngày',  n2,'/',t2,'/',nam2);
Writeln(' có',abs(ngaytd(n2,t2,nam2)-
                ngaytd(n1,t1,nam1))+1,'ngày. ');
Writeln('Nhấn Enter tiếp tục...');
Readln;
Writeln('Chờ 3 phút ...');
s1:=0;
s:=0;
For nam:=1800 to 1899 do
  For i:=ngaytd(1,1,nam) to ngaytd(31,12,nam) do
    Begin
      If (i mod 7=6) then s1:=s1+1;
      If (i mod 7=0) then s:=s+1;
    End;
Writeln('Trong thế kỉ XIX có',s1,' ngày thứ bảy và ',s,
        'ngày chủ nhật. ');
Writeln('Nhấn Enter tiếp tục...'); Readln;
Writeln('Nhập ngày tháng năm :');
Readln(n1,t1,nam1);
Writeln('Ngày ',n1,' tháng ',t1,' năm ',nam1,' là ',
        a{ngaytd(n1,t1,nam1) mod 7});
Writeln('Nhấn Enter kết thúc...');
Readln;
End.
```

§8. TỔNG ÔN TẬP CHƯƠNG TRÌNH

```

1. Program P4801;
   Uses Crt;
   Var N: longint;
   (*=====*)
   Function chuso(NN: longint): char;
   Var st: string[12];
       dem, M: longint;
   Begin
       dem:=0;
       M:=0;
       Repeat
           M:=M+2;
           Str(M,st);
           dem:=dem+length(st);
       Until dem>=NN;
       chuso:=st[length(st)-(dem-NN)];
   End;
   (*=====*)
   Begin
       Clrscr;
       Write('Nhập N(<=2147483647):'); Readln(N);
       Writeln('Chữ số thứ ',N,' của dãy vô hạn các số '
           +'nguyên không âm chẵn ');
       Write('0246810121416182022...là ',chuso(N));
       Readln;
   End.

```

2. Program P4802;

Uses Crt;

Var N: longint;

(*=====*)

Function chuso(NN: longint): char;

{Trả về chữ số thứ NN của dãy vô hạn các số nguyên
không âm lẻ 135791113151719212325.....}

Var st: string[10];

dem, M: longint;

Begin

dem:=0;

M:=-1;

Repeat

M:=M+2;

Str(M,st);

dem:=dem+length(st);

Until dem>=NN;

chuso:=st[length(st)-(dem-NN)];

End;

(*=====*)

Begin

Clrscr;

Write('Nhập N(<=2147483647):'); Readln(N);

Writeln('Chữ số thứ ',N, ' của dãy vô hạn các số
+nguyên không âm lẻ ');

Write('135791113151719212325...là:', chuso(N));

Readln;

End.

```

3. Program P4803;
   Uses Crt;
   Var N: longint;
   (*=====*)
   Function chuso(NN: longint): char;
   {Trả về chữ số thứ N của dãy vô hạn các số chính
   phương 1491625364981100121144... }
   Var Bp, M, dem: longint;
       s: string[10];
   Begin
       M:=0;
       dem:=0;
       Repeat
           inc(M);
           Bp:=M*M;
           str(Bp,s);
           inc(dem,length(s));
       Until dem>=NN;
       chuso:=s[length(s)-(dem-NN)];
   End;
   (*=====*)
   Begin
       Clrscr;
       Write('Nhập N='); Readln(N);
       Writeln('Chữ số thứ', N, 'của dãy vô hạn các số
       '+chính phương 1491625364981100121144...là:');
       Write(chuso(N));
       Readln;
   
```


End.

4. Program P4804;

Uses Crt;

Var N: longint;

(*=====*)

Function chuso(NN: longint): char;

{Trả về số thứ N của dãy vô hạn các số Fibonacci }

Var sod, soc, M, dem: longint;

s: string[12];

Begin

If NN=1 then chuso:='1'

Else

Begin

sod:=0; soc:=1;

dem:=1;

Repeat

M:=soc;

soc:=sod+soc; sod:=M;

str(soc,s);

inc(dem,length(s));

Until dem>=NN;

chuso:=s[length(s)-(dem-NN)];

End;

End;

(*=====*)

Begin

Clrscr;

```

Write('Nhập. ');
Repeat
    Write('N>0 : '); Readln(N);
    If N<=0 then Write('Nhập lại. ');
Until N>0;
Write('Chữ số thứ, N, ' của dãy vô hạn các số '
    + ' Fibonacci, là:', chuso(N));
Readln;
End.

```

5. Program P4805;

```

Uses Crt;
Var f: text;
    a,b: array[0..100] of integer;
    N,i: byte;
Begin
    Clrscr;
    Write('Nhập số N: '); Readln(n);
    assign(f, 'KQ. TXT');
    rewrite(f, n);
    a[0]:=1; b[0]:=2;
    Writeln(f, a[0], ' ', b[0]);
    For i:=1 to n do
        Begin
            a[i]:=a[i-1]+b[i-1];
            b[i]:=a[i-1]-b[i-1];
            Writeln(f, a[i], ' ', b[i]);
        End;

```

```
close(f);  
Write('Mở file KQ.TXT xem kết quả !');  
Write('Nhấn Enter để thoát..');  
Readln;  
End.
```

6. Program P4806;

```
Uses Crt;  
(*=====*)  
Function Init_st(Leng: byte): string;  
  {Trả về 1 xâu kí tự (gồm các kí tự từ 32 đến 127) và có  
  độ dài là Leng}  
  Var code, i: byte;  
      s: string;  
  Begin  
    randomize;  
    s:=""; {Xâu rỗng}  
    For i:=1 to leng do  
      Begin  
        Repeat  
          code:=random(128);  
        Until code>=32;  
        s:=concat(s,char(code));  
      End;  
    Init_st:=s;  
  End;  
(*=====*)  
Begin  
  Clrscr;
```

```

Writeln('Xâu ngẫu nhiên là ');
Writeln('St= ', Init_st(100));
Write('Nhấn Enter để thoát...');
Readln;
End.

```

7. Program P4707;

```

Uses Crt;
Var i, mbegin, mbegin: integer;
    s, pm, pm0, gap: byte;
    ph, ph0: byte;
Begin
    Clrscr;
    mbegin:=0;
    mbegin:=60*24;
    s:=0;
    pm0:=0;
    ph0:=0;
    For i:=mbegin to mbegin-1 do
        Begin
            pm:=i mod 60;
            ph:=(i div 12) mod 60;
            If (pm0<=ph0)and(pm>ph) then s:=s+1;
            pm0:=pm;
            ph0:=ph;
        End;
    If (pm0=ph0) then s:=s+1;
    Write('Từ 0h00 đến 24h00 kim giờ và kim phút gặp '

```

```
        + 'nhau ',s,' lần.');
```

```
    Readln;
```

```
End.
```

8. Program P4808;

```
Uses Crt;
```

```
Var  ci, ci0, ci1, pi, pi0, pi1: longint;
```

```
      qi, qi0, qi1, ri0, ri1, a, b: longint;
```

```
Begin
```

```
    Clrscr;
```

```
    Writeln('Tính p,q thoả mã n  $ap+bq = d$  với '
```

```
            + 'd =USCLN(a,b):');
```

```
    Write('Nhập số a: '); Readln(a);
```

```
    Write('Nhập số b: '); Readln(b);
```

```
    ci0:=a;
```

```
    ci1:=b;
```

```
    ri0:=0;
```

```
    pi0:= 1;
```

```
    qi0:=0;
```

```
    pi1:=0;
```

```
    qi1:=1;
```

```
    Repeat
```

```
        ci:=ci0 mod ci1;
```

```
        ri1:=ci0 div ci1;
```

```
        pi:=pi0-ri1*pi1;
```

```
        qi:=qi0-ri1*qi1;
```

```
        ci0:=ci1;
```

```
        ci1:=ci;
```

```
        ri0:=ri1;
```

```

        pi0:=pi1;
        pi1:=pi;
        qi1:=qi;
    Until (ci=0);
    WriteLn('USCLN(' ,a,',',b,') = ',ci0);
    Write('Ta có ', pi0, '*', a, '+', qi0, '*', b, ' = ', ci0);
    ReadLn;
End.

```

9. Program P4809;

```

Uses Crt;
Var ci, ci0, ci1, pi, pi0, pi1, ri0, ri1, a, b:longint;
Begin
    Clrscr;
    WriteLn('Tính B để  $A \cdot B \bmod p = 1$  với USCLN(A,p) '
    + '=1. ');
    Write('Nhập số A: '); ReadLn(a);
    Write('Nhập số B: '); ReadLn(b);
    ci0:=a;
    ci1:=b;
    ri0:=0;
    pi0:= 1;
    pi1:=0;
    Repeat
        ci:=ci0 mod ci1;
        ri1:=ci0 div ci1;
        pi:=pi0-ri1*pi1;
        ci0:=ci1;
        ci1:=ci;
        ri0:=ri1;
        pi0:=pi1;
    Until ci=0;

```

```

    pi1:=pi;
  Until (ci=0);
  If (ci0<>1) then Writeln(' USCLN(' ,a,',',b,') = ',ci0,' <> 1. ');
  Else
    Begin
      If (pi0<0) then pi0:=b+pi0;
      Writeln(' ',pi0,'*',a,' mod ',b,' = 1 . ');
    End;
  Readln;
End.

```

10. Program P4810;

```

  Uses Crt;
  Var f: text;
      n, m, i, j, q: longint;
      a, p: array [1..100] of longint;
  (*=====*)
  Function r(a, b: longint): longint;
  Var ci, ci0, ci1, pi, pi0, pi1, ri0, ri1: longint;
  Begin
    ci0:=a;
    ci1:=b;
    ri0:=0;
    pi0:= 1;
    pi1:=0;
    Repeat
      ci:=ci0 mod ci1;
      ri1:=ci0 div ci1;
      pi:=pi0-ri1*pi1;
      ci0:=ci1;
      ci1:=ci;
    Until ci=0;
  End;

```

```

        ri0:=ri1;
        pi0:=pi1;
        pi1:=pi;
    Until (ci=0);
    If (ci0<>1) then Writeln('USCLN(' , a, ', ', b, ') = ', ci0, '<>1.')
    Else If (pi0<0) then pi0:=b+pi0;
        r:=pi0;
End;
(*=====*)
Begin
    Clrscr;
    assign(f, 'in.inp');
    reset(f);
    Readln(f, m);
    Writeln('Tính N để N mod pi = ai với p1, p2, ...pm cho'
        +'trước (m =', m,') và USCLN(p1, p2, ...pm) = 1. ');
    Writeln('Nhập các số ai < pi : ');
    For i:=1 to m do
        Begin
            Readln(f, p[i]);
            Write('Nhập số[' , i, ']<p[' , i, ']=', p[i], ' : ');
            Readln(a[i]);
        End;
    n:=0;
    For i:=1 to m do
        Begin
            q:=1;
            For j:=1 to m do If (j<>i) then q:=q*p[j];
            n:=n+q*r(q, p[i])*a[i];
        End;

```



```

q:=1;
For j:=1 to m do q:=q*p[j];
n:=n mod q;
Writeln(' N = ', n);
Readln;
End.

```

11. Program P4811;

```

Uses      Crt;
Const    MaxN=1000;
Type     Khoang=1..MaxN;
Var      A, N, k: Khoang;
         next: char;
         SO: array[Khoang] of integer;
(*=====*)
Procedure input(Var Ai, Ni, ki: khoang);
Begin
  Writeln('Nhập A, k và N ( $1 \leq A \leq N$ )');
  Repeat
    Write('Số chữ số N='); Readln(Ni);
    If Ni<1 then Write('Nhập lại. ');
  Until Ni>=1;
  Repeat
    Write('Chữ số thứ K cần xóa k='); Readln(ki);
    If ki<1 then Write('Nhập lại. ');
  Until ki>=1;
  Repeat
    Write('Số bạn muốn máy xóa đi chậm nhất là');
    Write(' A='); Readln(Ai);
    If (Ai>Ni) or (Ai<1) then Write('Nhập lại. ');
  Until (1<=Ai) and (Ai<=Ni);

```

```

End;
(*=====*)
Procedure print(buoc, soxoa:integer; Var j: integer);
Begin
    textcolor(WHITE);
    Writeln(buoc, 'Xoá số ', soxoa);
    For j:=1 to N do
    Begin
        If SO[j]=0 then Textcolor(RED)
        ElseTextcolor(WHITE);
        If SO[j]<>0 then Write(SO[j]:5)
        Else Write('X':5);
    End;
    Writeln;
    delay(400);
End;
(*=====*)
Procedure play(A1, N1, k1: integer);
Var stop: boolean;
    count, buocxoa, diem, bk, i:integer;
Begin
    For i:=1 to N1 do SO[i]:=i;
    i:=0;
    buocxoa:=0;
    diem:=0;
    count:=0;
    stop:=false;
    Repeat
        If (i<=N1) then inc(i) else i:=1;
        If SO[i]<>0 then inc(count);

```

```

    If count=k1 then
        Begin
            count:=0;
            inc(diem);
            bk:=SO[i];
            inc(buocxoa);
            stop:=(SO[i]=A1);
            SO[i]:=0;  {Số bị xoá được gán bằng 0}
            print(buocxoa, bk);
        End;
    Until stop;
    textcolor(white);
    Writeln('Bạn chơi được ', diem, ' điểm');
End;
(*=====*)
Begin { Main Program }
    textbackground(black);
    window(1, 1, 80, 25);
    textcolor(white);
    next:='Y';
    While upcase(next)='Y' do
        Begin
            Clrscr;
            input(A, N, k);
            play(A, N, k);
            Write('Tiếp tục chơi(Y/N)?-');
            next:=readkey;
        End;
End.

```

```

12. Program P4812;
    Uses      Crt;
    Const
        Alpha: string[26]='ABCDEFGHIJKLMNOPQRSTUVWXYZ';
    Type      s10=string[10];
    Var  t1: set of char;
           st, ST1:s10;
           solan, max, min, saivt, dungvt, i, len:byte;

    (*=====*)
    Function init_st(k: byte): s10;
    {Trả về một xâu kí tự ngẫu nhiên có độ dài là k}
    Var  code, leng: byte;
           ss: s10;
    Begin
        randomize;
        Repeat
            leng:=random(k+1);
        Until (leng<>0);
        ss:=""; { Xâu rỗng }
        For i:=1 to leng do
            Begin
                code:=random(26);
                ss:=ss+ALPHA[code+1];
            End;
        init_st:=ss;
    End;

    (*=====*)
    Begin { Main Program }
        Clrscr;
        st:=init_st(10);

```

```
len:=length(st);
t1:=[];
solan:=0;
For i:=1 to len do t1:=t1+[st[i]];
Writeln( 'Độ dài của xâu ngẫu nhiên là :', len);
Repeat
    inc(solan);
    saivt:=0;
    dungvt:=0;
    gotoxy(1, 2);
    Write('Bạn nhập dòng chữ in hoa vào đây để thử:');
    clreol;
    Readln(STI);
    max:=length(STI);
    min:=len;
    If length(STI)<len then
        Begin
            min:=length(STI);
            max:=len;
        End;
    For i:=1 to max do
        If (STI[i] in t1)and(STI[i]<>st[i]) then inc(saivt);
    For i:=1 to min do
        If STI[i]=st[i] then inc(dungvt);
    gotoxy(1, 3);
    clreol;
    Writeln('Số chữ cái đúng nhưng sai vị trí là :',
    Writeln(saivt);
    gotoxy(1, 4);
    clreol;
```

```
    Writeln('Số chữ cái đúng vị trí là:', dungvt);  
    Until (STI=st);  
    gotoxy(1, 7);  
    Writeln('Số lần bạn thử là:', solan);  
    Readln;  
End.
```

13. Program P4713;

```
Uses    Crt;  
Const  
    Alpha: string[26]= 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';  
Type    s10=string[10];  
Var     t1: set of char;  
        st, STI: s10;  
        y, solan, sai, dung, i, len: byte;  
        ch: char;  
  
(*=====*)  
Function init_st(k:byte):s10;  
{Sinh một xâu ngẫu nhiên có độ dài là k}  
Var     code, leng: byte;  
        ss: s10;  
Begin  
    randomize;  
    Repeat  
        leng:=random(k+1);  
    Until leng<>0;  
    ss:="";  
    For i:=1 to leng do  
        Begin  
            code:=random(26);  
            ss:=ss+ALPHA[code+1];
```

```
End;
init_st:=ss;
End;
(*=====*)
Begin
  Clrscr;
  st:=init_st(10);
  len:=length(st);
  t1:=[];
  solan:=0;
  For i:=1 to len do t1:=t1+[st[i]];
  Writeln('Độ dài của xâu ngẫu nhiên là:', len);
  y:=6;
  Repeat
    inc(solan);
    sai:=0;
    dung:=0;
    Gotoxy(1, 2);
    clreol;
    Write('Bạn nhập dòng chữ in hoa vào để thử:');
    Readln(STI);
    For i:=1 to length(STI) do
      If STI[i] in t1 then inc(dung)
      Else inc(sai);
    Gotoxy(1, 3);
    clreol;
    Writeln('Số chữ cái đúng là:', dung);
    Gotoxy(1, 4);
    clreol;
    Writeln('Số chữ cái không đúng là:', sai);
```

```
Until (STI=st);  
Gotoxy(1, 7);  
Writeln('Số lần bạn thử là:', solan);  
Readln;  
End.
```

14. Program P4814;

```
Uses Crt;  
Const  st1=' nặng hơn.';  
        st2=' nhẹ hơn.';  
Var  i, kq1: integer;  
      kq2: string;  
      ch: char;  
(*=====*)  
Procedure kq(a: integer; b: string);  
Begin  
  kq1:=a;  
  kq2:=b;  
End;  
(*=====*)  
Procedure can(lan: integer; t1, t2, t3, t4: String; p1, p2,  
              p3, p4: String);  
Begin  
  Writeln('Lần cân thứ, lan, ':);  
  Writeln;  
  Writeln(' ', t1, ' ', t2, ' ', t3, ' ', t4, ' ', p1, ' ', p2, ' ', p3, ' ',  
    p4);  
  Writeln;  
  Write(' Bên nào nặng hơn? Trái(t)/Phải(p)/ Cân'  
    + 'bằng(c)');  
  Repeat
```



```

        ch:=readkey;
        ch:=upcase(ch);
        Until (ch in ['P', 'T', 'C']);
        Writeln(ch);
        Writeln('=====');
End;
(*=====*)
Procedure play;
Begin
    Writeln('Có 12 quả cân: 1 2 3 4 5 6 7 8 9 10 11 12');
    Writeln('Cho phép bạn chọn ra một quả cân nặng hơn'
        + 'hay nhẹ hơn những quả khác. ');
    can(1, '1', '2', '3', '4', '5', '6', '7', '8');
    If (ch='T') then {T}
    Begin
        can(2, '1', '2', '5', '', '3', '4', '6', '');
        If (ch='T') then {TT}
        Begin
            can(3, '1', '6', '', '', '7', '8', '', '');
            If ch='T' then kq(1, st1); {TTT}
            If ch='P' then kq(6, st2); {TTP}
            If ch='C' then kq(2, st1); {TTC}
        End
    Else If (ch='P') then {TP}
    Begin
        can(3, '3', '5', '', '', '7', '8', '', '');
        If ch='T' then kq(3, st1); {TPT}
        If ch='P' then kq(5, st2); {TPP}
        If ch='C' then kq(4, st1); {TPC}
    End
End

```

```

Else If (ch='C') then           {TC}
Begin
    can(3, '7', ' ', ' ', ' ', ' ', '8', ' ', ' ');
    If ch='T' then kq(8, st2);   {TCT}
    If ch='P' then kq(7, st2);   {TCP}
    If ch='C' then
        Begin
            WriteLn('Trả lời sail!');
            kq2:=st2;
        End;
    End;
End
Else If (ch='P') then           {P}
Begin
    can(2, '5', '6', '1', ' ', '7', '8', '2', ' ');
    If (ch='T') then             {PT}
        Begin
            can(3, '5', '2', ' ', ' ', '3', '4', ' ', ' ');
            If ch='T' then kq(5, st1);
            If ch='P' then kq(2, st2);
            If ch='C' then kq(6, st1);
        End
    Else If (ch='P') then         {PP}
        Begin
            can(3, '7', '1', ' ', ' ', '3', '4', ' ', ' ');
            If ch='T' then kq(7, st1);
            If ch='P' then kq(1, st2);
            If ch='C' then kq(8, st1);
        End
    Else If (ch='C') then         {PC}

```

```

Begin
    can(3, '3', '', '', '', '4', '', '');
    If ch='T' then kq(4, st2);
    If ch='P' then kq(3, st2);
    If ch='C' then
        Begin
            Writeln('Trả lời sai !');
            kq2:=st2;
        End;
    End;
End
Else If (ch='C') then {C}
    Begin
        can(2, '9', '10', '11', '', '1', '2', '3', '');
        If (ch='T') then {CT}
            Begin
                can(3, '9', '', '', '10', '', '', '');
                If (ch='T') then kq(9, st1);
                If (ch='P') then kq(10, st1);
                If (ch='C') then kq(11, st1);
            End
        Else If (ch='P') then {CP}
            Begin
                can(3, '9', '', '', '10', '', '', '');
                If (ch='T') then kq(10, st2);
                If (ch='P') then kq(9, st2);
                If (ch='C') then kq(11, st2);
            End
    End

```

```

Else If (ch='C') then           {CC}
    Begin
        can(3, '12', '', '', '', '1', '', '', '');
        If (ch='T') then kq(12, st1);
        If (ch='P') then kq(12, st2);
        If (ch='C') then Writeln('Trả lời sai!');
        kq1:=12;
    End;
End;

End;
(*=====*)
Begin
    Clrscr;
    play;
    Writeln(' Quả thứ', kq1, kq2);
    Writeln(' Nhấn Enter kết thúc...');
    Readln;
End.

```

15. Program P4815;

```

Uses Crt;
Const Max=10000;
Var fin, fout: text;
    x: array[1..Max] of word;
    N, m: word;
(*=====*)
Procedure Motep(Var ff: text; tentep: string; md: byte);
{Mở một tệp có tên là tentep: md=0 mở để đọc, md=1 mở để ghi}

```

```

Begin
    assign(ff, tentep);
    {$I-}
    If md=0 then reset(ff) Else If md=1 then rewrite(ff);
    {$I+}
    If ioresult<>0 then
        Begin
            WriteLn(Không tìm thấy hoặc không mở được'
                    +'tệp' +tentep);
            Write('Nhấn Enter để thoát...');
            ReadLn;
            halt(0);
        End;
    End;
    (*=====*)
    Procedure ghi_vao_tep(Var ff:text);
    Var i1:byte;
    Begin
        For i1:=1 to m do Write(ff, x[i1]:5);
        WriteLn(ff); {Xuống một dòng}
    End;
    (*=====*)
    Procedure Giai_ph_tr(N1, M1: word);
    Var d, T, dem, i, idl, bk: word;
        kt: boolean;
    Begin
        If M1>=2 then
            Begin

```

```

bk:=1;
x[bk]:=0;
dem:=0; {Đếm số nghiệm}
idl:=M1-1;
d:=0;
Repeat
  T:=0;
  kt:=true;
  For i:=1 to bk do T:=T+x[i];
  While (i<M1)and kt do
    {Tìm một nghiệm thỏa mãn  $x_1+x_2+\dots+x_n=N$ }
    Begin
      inc(i);
      If i<=m-1 then x[i]:=x[i-1]+1
      Else x[i]:=N1-T;
      inc(T, x[i]);
      kt:=(x[i]>x[i-1]);
    End;
  inc(d);
  If kt then {Đã tìm được 1 nghiệm}
  Begin
    ghi_vao_tep(fout);
    inc(dem);
    While (x[M1]-x[M1-1]>2) and (x[idl]<x[idl+1]) do
      Begin
        inc(dem);
        inc(x[i-1]);
        dec(x[i]);

```

```

        ghi_vao_tep(fout);
    End;
End
Else dec(idl);
    bk:=idl;
    inc(x[idl]);
Until (idl=0)or((d=1)and not kt);
Writeln(fout, 'Phương trình X1+X2+....+X', M, '= ', N);
If dem=0 then Write(fout, 'Không có nghiệm nào')
Else    Write(fout, 'có ', dem, ' nghiệm ở trên');
End
Else
    If M1=1 then
        Writeln(fout, 'Phương trình X1+X2+....+Xm=N '
            +'có 1 nghiệm x1=', N1)
    Else
        Writeln(fout, 'đây không phải là phương trình');
Writeln;
End;
(*=====*)
Begin { Main Program }
    Clrscr;
    Motep(fin, 'IN.TXT', 0);
    Motep(fout, 'OUT.TXT', 1);
    Readln(fin); {Nhảy qua dòng tiêu đề}
    Readln(fin, N, m);
    close(fin);
    Writeln('Giải phương trình X1+X2+....+Xm=N ');

```

```
WriteLn('Đợi và xem kết quả trong tệp OUT.TXT...');
Giai_ph_tr(N, m);
close(fout);
Write('Nhấn Enter để thoát..');
ReadLn;
End.
```

16. Program P4816;

```
Uses Crt;
Const Max=500;
Var  fout: text;
      x: array[1..Max] of word;
      N: word;

(*=====*)
Procedure Write_to_file(mm: integer; Var ff: text);
Var i1: byte;
Begin
  For i1:=1 to mm do Write(ff, x[i1]:5);
  WriteLn(ff); { Xuống 1 dòng}
End;

(*=====*)
Procedure Giai_ph_tr(N1: word);
Var  m1, T, count, i, idl: word;
      mk, kt: boolean;
Begin
  T:=0;
  m1:=1;
  x[m1]:=0;
```



```

kt:=false;
While not kt do
  Begin
    If (T+x[m1]<=N1)and(T+2*x[m1]+1>N1) then
      x[m1]:=N1-T;
    inc(T, x[m1]);
    kt:=(T=N1);
    inc(m1);
    x[m1]:=x[m1-1]+1;
  End; { of While ... }
dec(m1);
idl:=m1-1;
mk:=false;
count:=0; {Đếm số nghiệm}
Repeat
  If mk then
    Begin
      T:=0;
      i:=1;
      kt:=true;
      If idl<>0 then For i:=1 to idl do T:=T+x[i]
      Else x[1]:=0;
      While (i<m1)and kt do
        Begin
          inc(i);
          If i<=m1-1 then x[i]:=x[i-1]+1
          Else x[i]:=N1-T;
          inc(T, x[i]);
        End;
    End;
  Until kt;
  count:=count+1;
  mk:=not kt;
Until count>0;

```

```
        kt:=(x[i]>x[i-1]);
    End;
End;
If kt then {Đã tìm được 1 nghiệm}
Begin
    Write_to_file(m1, fout)
    inc(count);
    While (x[m1]-x[m1-1]>2) do
        Begin
            inc(count);
            inc(x[m1-1]);
            dec(x[m1]);
            Write_to_file(m1, fout);
        End;
    End
    Else dec(idl);
    inc(x[idl]);
    mk:=true;
    If idl=0 then dec(m1);
Until (m1=1);
Writeln(fout, N1:5);
Writeln(fout, 'Với N=', N1, ' thì ');
Writeln(fout, 'Phương trình  $X_1+X_2+...+X_m+...=N$ ');
If count=0 then Write(fout, 'Không có nghiệm nào')
Else Write(fout, 'có ', count+1, ' nghiệm ở trên');
Writeln;
End;
(*=====*)
```

```
Begin { Main Program }
Clrscr;
  assign(fout, 'OUT.TXT');
  {$I-}
  reWrite(fout);
  {$I+}
  If ioresult<>0 then
    Begin
      Writeln('Không mở được tệp OUT.TXT. ');
      Writein('Nhấn Enter để thoát...');
      Readln;
      halt(0);
    End;
  Writeln('Giải phương trình  $X_1+X_2+...+X_m+...=N$  ');
  Writeln('Thoả mã n  $X_1<X_2<.....<X_m<...$  ');
  Write('N='); Readln(N);
  Writeln('Đợi và xem kết quả trong tệp OUT.TXT... ');
  Giai_ph_tr(N);
  close(fout);
  Write('Nhấn Enter để thoát.. ');
  Readln;
End.
```

17. Program P4817;

```
Uses Crt;
Const Maxint=10000;
  Chuso=['0'..'9'];
  mcs: array['0'..'9'] of byte=(0, 1, 2, 3, 4, 5, 6, 7, 8, 9);
Type SNL=array[0..Maxint] of 0..9;
```

```

Var  A, C: SNL;
      b: longint;
      i, scsa, scsc: word;
(*=====*)
Procedure Input(Var SS: SNL; Var Nss: word);
{Nhập số nguyên lớn SS có Nss chữ số}
Var ch: char;
Begin
  Nss:=0;
  Write('Nhập số a=');
  Repeat
    ch:=readkey;
  Until ch in ( Chuso+['-', '+']);
  Write(ch);
  case ch of
    '+':SS[0]:=1;
    '-':SS[0]:=0;
    '0'..'9': Begin
      SS[0]:=1;
      SS[1]:=mcs[ch];
      Nss:=1;
    End;
  End;
Repeat
  ch:=readkey;
  If ch in Chuso then
    Begin
      Write(ch);
      inc(Nss);
    End;
  End;
End;

```

```

        SS[Nss]:=mcs[ch];
    End;
    Until ch=#13;
    WriteLn;
End;
(*=====*)
Procedure Mul(SB: SNL; numb: word; X: longint; Var SC:
    SNL; Var numc: word);
{Nhân số nguyên lớn SB có numb chữ số với số nguyên X}
{Tích được lưu vào SC, số chữ số của tích được lưu vào numc}
Var nh: longint;
Begin
    numc:=0;
    nh:=0;
    For i:=numb downto 1 do
        Begin
            inc(numc);
            nh:=SB[i]*abs(X)+nh;
            SC[numc]:=nh mod 10;
            nh:=nh div 10;
        End;
    While nh<>0 do
        Begin
            inc(numc);
            SC[numc]:=nh mod 10;
            nh:=nh div 10;
        End;
End;

```

```
If ((X<0)and(SB[0]=0))or((X>=0)and(SB[0]=1)) then
    SC[0]:=1
```

```
Else SC[0]:=0;
```

```
For i:=1 to (numc+1) div 2 do
```

```
    Begin
```

```
        nh:=SC[i];
```

```
        SC[i]:=SC[numc-i+1];
```

```
        SC[numc-i+1]:=nh;
```

```
    End;
```

```
End;
```

```
(*=====*)
```

```
Procedure print(SS: SNL; nums: word);
```

```
{In số nguyên lớn SS có nums chữ số}
```

```
Begin
```

```
    If SS[0]=0 then Write('-') { Số âm}
```

```
    Else Write('+'); { Số dương}
```

```
    For i:=1 to nums do Write(SS[i]);
```

```
End;
```

```
(*=====*)
```

```
Begin
```

```
    Clrscr;
```

```
    Input(a, scsa);
```

```
    Write('Nhập số b='); Readln(b);
```

```
    Mul(A, scsa, b, C, scsc);
```

```
    Write('a*b=');
```

```
    print(C, scsc);
```

```
    Readln;
```

```
End.
```

18. Program P4818;

```
Uses Crt;
```

```

Const Maxint=30000;
Type SNL=array[1..Maxint] of 0..9;
Var  A: SNL;
      i, numa, N: integer;
(*=====*)
Procedure Mul(X: integer; Var A1: SNL;
              Var num1: integer);
{Nhân số tự nhiên X với số lớn A1 kết quả để trong A1}
Var  nh, id: longint;
Begin
  nh:=0;
  For id:=1 to num1 do
    Begin
      nh:=A1[id]*X+nh;
      A1[id]:=nh mod 10;
      nh:=nh div 10;
    End;
  While nh<>0 do
    Begin
      inc(num1);
      A1[num1]:=nh mod 10;
      nh:=nh div 10;
    End;
End;
(*=====*)
Procedure Giai_Thua(Nn: integer; Var GT: SNL; Var
                   Nn1: integer);
{ Tính giai thừa của số Nn kết quả được lưu trong GT }
Var  tg: byte;
      k, it: integer;

```

```

Begin
  Nn1:=1;
  if Nn in [0, 1] then GT[Nn1]:=1
  Else
    Begin {Tính giai thừa của số Nn>1 }
      GT[Nn1]:=1;
      k:=2;
      Repeat
        Mul(k, GT, Nn1);
        inc(k);
      Until k>Nn;
      For it:=1 to (Nn1+1) div 2 do
        Begin
          tg:=GT[it];
          GT[it]:=GT[Nn1-it+1];
          GT[Nn1-it+1]:=tg;
        End;
      End;
    End;
  End;
  (*=====*)
  Begin
    Clrscr;
    Writeln('Tính N! với');
    Repeat
      Write('N='); Readln(N);
      If N<0 then Writeln('Nhập N>=0. ');
    Until N>=0;
    Writeln('Bạn hãy đợi kết quả...');
    Giai_Thua(N, A, numa);
    Clrscr;
  End;

```



```

Write('Ta có ', N, '!=');
For i:=1 to numa do Write(A[i]);
Writeln;
Write(N, '!= có ', numa, ' chữ số');
Readln;
End.

```

19. Program P4819;

```

Uses Crt;
Const Maxint=10000;
      tapcs=['0'..'9'];
      mcs: array['0'..'9'] of byte=(0, 1, 2, 3, 4, 5, 6, 7, 8, 9);
Type  SNL=array[1..Maxint] of 0..9;
Var   A, B: SNL;
      i, N, numa, numb: word;
      (*=====*)
Procedure Input(Var SS: SNL; Var num: word);
{Nhập số nguyên lớn SS có num chữ số}
Var ch: char;
Begin
  num:=0;
  Write('Nhập số nguyên a=');
  Repeat
    ch:=readkey;
    If ch in tapcs then
      Begin
        Write(ch);
        inc(num);
        SS[num]:=mcs[ch];
      End;
  Until ch=#13;

```

```
        Writeln;
    End;
    (*=====*)
    Procedure DIVS(A: SNL; numA, B: word; Var C: SNL;
        Var numC: word);
    { Tính C= A div B }
    Var T, id: longint;
    Begin
        id:=0;
        numC:=0;
        T:=0;
        Repeat
            Repeat
                inc(id);
                T:=T*10+a[id];
            Until ((T div B)>=1)or(T=0)or(id=numA);
            inc(numC);
            C[numC]:=T div B;
            T:=T mod B;
        Until id=numA;
    End;
    (*=====*)
    Begin
        Clrscr;
        Input(A, numa);
        Repeat
            Write('Nhập số tự nhiên N='); Readln(N);
        Until N>0;
        DIVS(A, numa, N, B, numB);
        Write('A div N=');
```

```
    For i:=1 to numB do Write(B[i]);  
    Readln;  
End.
```

```
20. Program P4820;  
Uses Crt;  
Const Maxint=10000;  
      tapcs=['0'..'9'];  
      mcs: array['0'..'9'] of byte=(0, 1, 2, 3, 4, 5, 6, 7, 8, 9);  
Type  SNL=array[1..Maxint] of 0..9;  
Var   A: SNL;  
      NumA: word;  
      s: array[1..100] of string[80];  
(*=====*)  
Procedure Input(Var SS: SNL; Var num: word);  
{Nhập số nguyên lớn SS có num chữ số}  
Var ch: char;  
Begin  
    num:=0;  
    Write('A=');  
    Repeat  
        ch:=readkey;  
        If ch in tapcs then  
            Begin  
                Write(ch);  
                inc(num);  
                SS[num]:=mcs[ch];  
            End;  
    Until ch=#13;  
    Writeln;  
End;
```

```

(*=====*)
Procedure Bo_so_0(Var SB: SNL; Var numSB: word);
{ Cắt bỏ các số 0 ở đầu của số SB }
Var id: word;
Begin
  While (numSB>1)and(SB[1]=0) do
    Begin
      For id:=1 to numSB-1 do SB[id]:=SB[id+1];
      dec(numSB);
    End;
End;
(*=====*)
Procedure Doi_nhiphan(AA: SNL; na: word);
{Đổi số AA có na chữ số ở dạng thập phân sang dạng nhị
phân}
Var T, i, j, k: word;
Begin
  Write('A=');
  k:=1;
  s[k]:=""; {Xâu rỗng}
  Repeat
    j:=0;
    i:=0;
    T:=0;
    Repeat
      Repeat
        inc(i);
        T:=T*10+AA[i];
      Until ((T div 2)>=0)or(i=na);
      inc(j);

```

```
        AA[j]:=T div 2;
        T:=T mod 2;
    Until i=na;
    Bo_so_0(AA, j);
    na:=j;
    If length(s[k])<80 then s[k]:=char(T+48)+s[k]
    Else inc(k);
    Until (j=1)and(AA[j]=0);
    For i:=k downto 1 do Write(s[i]);
End;
(*=====*)
Begin { Main Program }
Clrscr;
    Writeln('Nhập số lớn A');
    Input(A, numA);
    Writeln('Đổi sang dạng nhị phân là');
    Doi_nhiphan(A, numA);
    Writeln;
    Write('Nhấn Enter để kết thúc...');
    Readln;
End.
```

Mục lục

Phần I: Bài tập	3
§ 1: Ôn tập: Dữ liệu và thuật toán trong Pascal	3
§ 2: Khái niệm thủ tục	9
§ 3: Tham số của thủ tục	12
§ 4: Khái niệm hàm	18
§ 5: Biến cục bộ, biến tổng thể. Truyền tham biến cho thủ tục / hàm	21
§ 6: Làm việc với file trong Pascal	28
§ 7: Ôn luyện	35
§ 8: Tổng ôn tập chương trình	40
Phần II: Hướng dẫn	46
§ 1: Ôn tập: Dữ liệu và thuật toán trong Pascal	46
§ 2: Khái niệm thủ tục	66
§ 3: Tham số của thủ tục	86
§ 4: Khái niệm hàm	104
§ 5: Biến cục bộ, biến tổng thể. Truyền tham biến cho thủ tục / hàm	121
§ 6: Làm việc với file trong Pascal	143
§ 7: Ôn luyện	171
§ 8: Tổng ôn tập chương trình	201

Chịu trách nhiệm xuất bản:
Giám đốc PHẠM VĂN AN
Tổng biên tập NGUYỄN NHƯ Ý

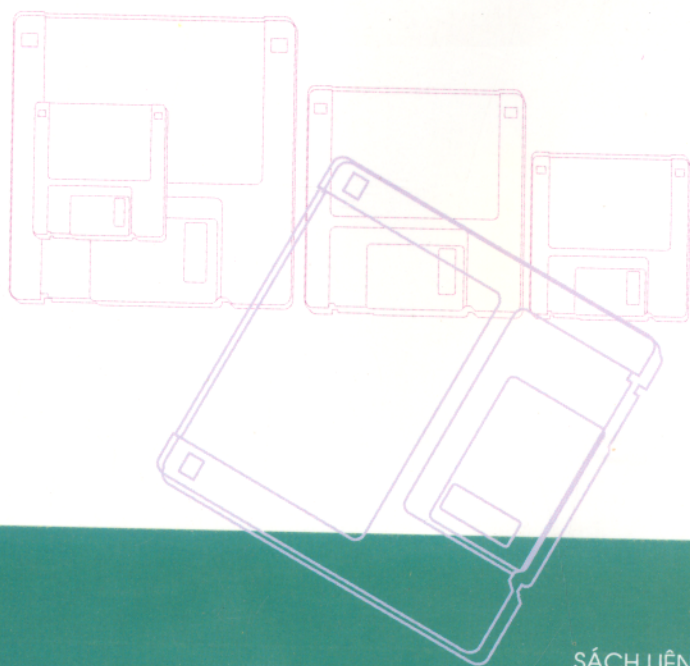
Biên tập nội dung:
NGÔ ÁNH TUYẾT

TỰ HỌC LẬP TRÌNH PASCAL - TẬP 4

In 1000 cuốn, khổ 14,5 × 20,5 tại Công ty khảo sát và xây dựng.
Số giấy chấp nhận KHXB số 366/CXB. Cục XB kí ngày
3/5/1999. In xong và nộp lưu chiểu tháng 8 năm 1999.

TỰ HỌC LẬP TRÌNH

PASCAL



SÁCH LIÊN KẾT XUẤT BẢN
GIÁ: 13500đ