

## Bài thực hành 6

# SỬ DỤNG LỆNH LẶP WHILE...DO

### 1. Mục đích, yêu cầu

- Viết chương trình Pascal sử dụng câu lệnh lặp với số lần chưa biết trước.
- Rèn luyện khả năng đọc chương trình, tìm hiểu tác dụng của các câu lệnh.

### 2. Nội dung

**BÀI 1.** Viết chương trình sử dụng lệnh lặp **while...do** để tính trung bình  $n$  số thực  $x_1, x_2, x_3, \dots, x_n$ . Các số  $n$  và  $x_1, x_2, x_3, \dots, x_n$  được nhập vào từ bàn phím.

**Ý tưởng:** Sử dụng một biến đếm và lệnh lặp **while...do** để nhập và cộng dần các số vào một biến kiểu số thực cho đến khi nhập đủ  $n$  số.

a) Mô tả thuật toán của chương trình, các biến dự định sẽ sử dụng và kiểu của chúng.

b) Gõ chương trình sau đây và lưu chương trình với tên **Tinh\_TB** :

```
program Tinh_Trung_binh;
uses crt;
var n, dem : Integer;
    x, TB : real;
begin
  clrscr;
  dem:= 0; TB:= 0;
  write('Nhap so cac so can tinh n = '); readln(n);
  while dem < n do
    begin
      dem:= dem + 1;
      write('Nhap so thu ', dem, ' = '); readln(x);
      TB:= TB + x;
    end;
  TB:= TB/n;
  writeln('Trung binh cua ', n, ' so la = ', TB:10:3);
```

```
writeln('Nhan Enter de thoat ...');
readln
```

**end.**

c) Đọc và tìm hiểu ý nghĩa của từng câu lệnh. Dịch chương trình và sửa lỗi, nếu có. Chạy chương trình với các bộ dữ liệu được gõ từ bàn phím và kiểm tra kết quả nhận được.

d) Viết lại chương trình bằng cách sử dụng câu lệnh **for...do** thay cho câu lệnh **while...do**.

**BÀI 2.** Tìm hiểu chương trình nhận biết một số tự nhiên  $N$  được nhập vào từ bàn phím có phải là số nguyên tố hay không.

**Ý tưởng:** Kiểm tra lần lượt  $N$  có chia hết cho các số tự nhiên  $2 \leq i \leq N - 1$  hay không. Kiểm tra tính chia hết bằng phép chia lấy phần dư (*mod*).

a) Đọc và tìm hiểu ý nghĩa của từng câu lệnh trong chương trình sau đây:

```
uses Crt;
var n, i: integer;
begin
  clrscr;
  write('Nhap vao mot so nguyen: '); readln(n);
  If n <= 1 then writeln(n, ' khong la so nguyen to')
  else
    begin
      i:= 2;
      while (n mod i<>0) do i:= i + 1;
      if i = n then writeln(n, ' la so nguyen to !')
      else writeln(n, ' khong phai la so nguyen to!');
    end;
  readln
end.
```

b) Gõ, dịch và chạy thử chương trình với một vài độ chính xác khác nhau.

## TỔNG KẾT

Câu lệnh lặp **while...do** có dạng

```
while <điều kiện> do <câu lệnh>;
```



### Đọc thêm 2. Tính gần đúng số $\pi$ với độ chính xác cho trước

Trong bài đọc thêm 1 "Tính gần đúng số  $\pi$ ", ta đã sử dụng công thức

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{1}{2n-1} - \frac{1}{2n+1} + \dots$$

để viết chương trình tính số  $\pi$  với  $n$  số hạng cho trước.

Sử dụng lệnh **while...do**, ta còn có thể viết chương trình để tính gần đúng số  $\pi$  với độ chính xác theo yêu cầu (được nhập từ bàn phím):

```
uses crt;  
var SoPi, saiso, dochinhxac: real;  
    n, i, dau: integer;  
begin  
    clrscr;  
    write('Hay cho sai so de tinh gan dung so Pi =');  
    readln(saiso);  
    SoPi:= 0; dochinhxac:= 3; i:= 0; dau:= -1;  
    while dochinhxac >= saiso do  
        begin dau:= -dau; SoPi:= SoPi + dau*1/(2*i + 1);  
            dochinhxac:= Abs(4*SoPi - Pi); i:= i + 1 end;  
            {Pi la ham chuan}  
    writeln('So Pi gan bang ', SoPi*4);  
    readln  
end.
```

**Chú thích.** Chương trình trên đã sử dụng hàm chuẩn **Abs** của Pascal. Hàm **Abs** cho kết quả là giá trị tuyệt đối của một số, tức **Abs(x)** cho giá trị  $x$ , nếu  $x \geq 0$ , ngược lại **Abs(x)** cho kết quả  $-x$ .