

BÀI 7

CÂU LỆNH LẶP

1. Các công việc phải thực hiện nhiều lần

Trong cuộc sống hàng ngày, nhiều hoạt động được thực hiện lặp đi lặp lại nhiều lần.

Có những hoạt động mà chúng ta thường thực hiện lặp lại với một số lần nhất định và biết trước, chẳng hạn đánh răng mỗi ngày hai lần, mỗi ngày tắm một lần,... Chúng ta còn lặp lại những công việc với số lần không thể xác định trước: học cho đến khi thuộc bài, nhặt từng cọng rau cho đến khi xong,...

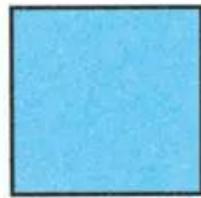
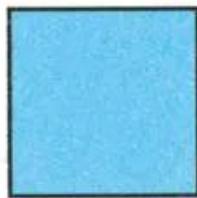
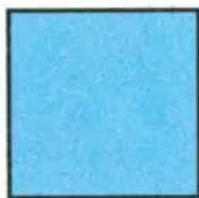
Khi viết chương trình máy tính cũng vậy. Để chỉ dẫn cho máy tính thực hiện đúng công việc, trong nhiều trường hợp ta cũng cần phải viết lặp lại nhiều câu lệnh thực hiện một phép tính nhất định.

2. Câu lệnh lặp - một lệnh thay cho nhiều lệnh

Ví dụ 1. Giả sử ta cần vẽ ba hình vuông có cạnh 1 đơn vị như hình 33. Mỗi hình vuông là ảnh dịch chuyển của hình bên trái nó một khoảng cách 2 đơn vị. Do đó, ta chỉ cần lặp lại thao tác vẽ hình vuông ba lần. Việc vẽ hình có thể thực hiện được bằng thuật toán sau đây:

Bước 1. Vẽ hình vuông (vẽ liên tiếp bốn cạnh và trở về đỉnh ban đầu).

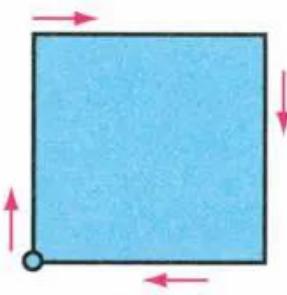
Bước 2. Nếu số hình vuông đã vẽ được ít hơn 3, di chuyển bút vẽ về bên phải 2 đơn vị và trở lại bước 1; ngược lại, kết thúc thuật toán.



Hình 33

Riêng với bài toán vẽ một hình vuông (h. 34), thao tác chính là vẽ bốn cạnh bằng nhau, hay lặp lại bốn lần thao tác vẽ một đoạn thẳng. Sau mỗi lần

vẽ đoạn thẳng, thước kẻ được quay một góc 90° sang phải tại vị trí của bút vẽ. Thuật toán sau đây sẽ mô tả các bước để vẽ hình vuông:



Hình 34

Bước 1. $k \leftarrow 0$ (k là số đoạn thẳng đã vẽ được).

Bước 2. $k \leftarrow k + 1$. Vẽ đoạn thẳng 1 đơn vị độ dài và quay thước 90° sang phải.

Bước 3. Nếu $k < 4$ thì trở lại bước 2; Ngược lại, kết thúc thuật toán.

Lưu ý rằng, biến k được sử dụng như là biến đếm để ghi lại số cạnh đã vẽ được.

Ví dụ 2. Giả sử cần tính tổng của 100 số tự nhiên đầu tiên, tức là tính:

$$S = 1 + 2 + 3 + \dots + 100.$$

Hoạt động chính khi giải bài toán này là thực hiện phép cộng. Thuật toán trong ví dụ 3, bài 5 đã mô tả việc thực hiện lặp lại phép cộng 100 lần.

Cách mô tả các hoạt động lặp trong thuật toán như trong ví dụ trên được gọi là **cấu trúc lặp**.

Mỗi ngôn ngữ lập trình đều có “cách” để chỉ thị cho máy tính thực hiện cấu trúc lặp với một câu lệnh. Đó là các **câu lệnh lặp**.

3. Ví dụ về câu lệnh lặp

Các ngôn ngữ lập trình thường có nhiều dạng câu lệnh lặp. Câu lệnh lặp thường gặp trong Pascal có dạng:

for *<biến đếm>* := *<giá trị đầu>* **to** *<giá trị cuối>* **do** *<câu lệnh>*;

Trong đó, **for**, **to**, **do** là các từ khóa, **biến đếm** là biến kiểu nguyên, giá trị đầu và giá trị cuối là các giá trị nguyên.

Câu lệnh lặp sẽ thực hiện **câu lệnh** nhiều lần, mỗi lần là một vòng lặp. Số vòng lặp là biết trước và bằng

$$\text{giá trị cuối} - \text{giá trị đầu} + 1.$$

Khi thực hiện, ban đầu biến đếm sẽ nhận giá trị là **giá trị đầu**, sau mỗi vòng lặp, biến đếm được tự động tăng thêm một đơn vị cho đến khi bằng **giá trị cuối**.

Ví dụ 3. Chương trình sau sẽ in ra màn hình thứ tự lần lặp:

```
program Lap;
var i: Integer;
begin
  for i:= 1 to 10 do
    writeln('Day la lan lap thu ',i);
  Readln
end.
```

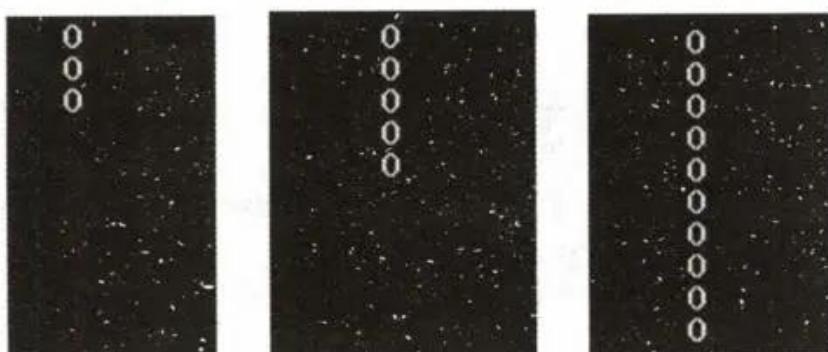
Ví dụ 4. Để in một chữ “O” trên màn hình, ta có thể sử dụng lệnh:

```
writeln('O');
```

Nếu muốn viết chương trình ghi nhận các vị trí của một quả trứng rơi từ trên cao xuống, ta có thể lặp lại lệnh trên nhiều lần (ví dụ, 20 lần) như trong chương trình sau:

```
uses crt;
var i: integer;
begin
  clrscr;
  for i:= 1 to 20 do
    begin writeln('O'); delay(100) end;
  Readln
end.
```

Dịch và chạy chương trình này, ta sẽ thấy kết quả như ở hình 35 dưới đây:



Hình 35

Lưu ý: Trong ví dụ 4, các câu lệnh đơn giản `writeln('O')` và `delay(100)` được đặt trong hai từ khoá `begin` và `end` để tạo thành một câu lệnh ghép trong Pascal. Từ đây về sau, khi nói câu lệnh, ta có thể hiểu đó là câu lệnh đơn hoặc câu lệnh ghép.

Trong thực tế, để có mười kết quả, cho dù là giống nhau hay khác nhau, chúng ta phải thực hiện hoạt động mười lần (cùng một hoạt động giống nhau hay các hoạt động khác nhau). Máy tính thực hiện công việc xử lý thông tin thay cho con người và cũng phải thực hiện ngần ấy hoạt động. Do vậy câu lệnh lặp giúp giảm nhẹ công sức viết chương trình máy tính.

4. Tính tổng và tích bằng câu lệnh lặp

Ví dụ 5. Chương trình sau đây sẽ tính tổng của N số tự nhiên đầu tiên, với N là số tự nhiên được nhập vào từ bàn phím (xem ví dụ 2).

```
program Tinh_tong;
var N, i: integer;
    S: longint;
begin
    write('Nhập số N = '); readln(N);
    S:=0;
    for i:= 1 to N do S:= S + i;
    writeln('Tổng của ',N,' số tự nhiên đầu tiên S = ',S);
    Readln
end.
```

Lưu ý. Vì với N lớn, tổng của N số tự nhiên đầu tiên có thể rất lớn nên trong chương trình trên ta sử dụng một kiểu dữ liệu mới của Pascal, kiểu `longint` (được khai báo cho biến S). Đây cũng là kiểu số nguyên, nhưng có thể lưu các số nguyên trong phạm vi từ -2^{31} đến $2^{31}-1$, lớn hơn nhiều so với kiểu `integer` (chỉ từ -2^{15} đến $2^{15}-1$).

Ví dụ 6. Ta ký hiệu $N!$ là tích N số tự nhiên đầu tiên:

$$N! = 1.2.3...N$$

Dưới đây là chương trình tính $N!$ với N là số tự nhiên được nhập vào từ bàn phím. Chương trình sử dụng một câu lệnh lặp `for...do`:

```

program Tinh_Giai_thua;
var N, i: Integer;
P: longint;
begin
  write('N = '); readln(N);
  P:= 1;
  for i:= 1 to N do P:=P*i;
  writeln(N, '! = ', P);
  Readln
end.

```

Lưu ý. Vì $N!$ là số rất lớn so với N, một lần nữa cần lưu ý khai báo biến chứa giá trị của nó đủ lớn.

GHI NHÓ

1. Cấu trúc lặp được sử dụng để chỉ thị cho máy tính thực hiện lặp lại một vài hoạt động nào đó cho đến khi một điều kiện nào đó được thoả mãn.
2. Mọi ngôn ngữ lập trình đều có các câu lệnh lặp để thể hiện cấu trúc lặp.
3. Ngôn ngữ Pascal thể hiện cấu trúc lặp với số lần lặp cho trước bằng câu lệnh **for...do**.

Câu hỏi và bài tập

1. Cho một vài ví dụ về hoạt động được thực hiện lặp lại trong cuộc sống hằng ngày.
2. Hãy cho biết tác dụng của câu lệnh lặp với số lần biết trước.
3. Khi thực hiện câu lệnh lặp, máy tính kiểm tra một điều kiện. Với lệnh lặp **for <biến đếm>:= <giá trị đầu> to <giá trị cuối> do <câu lệnh>;** của Pascal, điều kiện cần phải kiểm tra là gì?

4. Sau khi thực hiện đoạn chương trình sau, giá trị của biến j bằng bao nhiêu?

```
j:= 0;  
for i:= 0 to 5 do j:= j + 2;
```

5. Các câu lệnh Pascal sau có hợp lệ không, vì sao?

- a) `for i:= 100 to 1 do writeln('A');`
- b) `for i:= 1.5 to 10.5 do writeln('A');`
- c) `for i= 1 to 10 do writeln('A');`
- d) `for i:=1 to 10 do; writeln('A');`
- e) `var x :real; begin for x:= 1 to 10 do writeln('A'); end.`

6. Hãy mô tả thuật toán để tính tổng sau đây :

$$A = \frac{1}{1.3} + \frac{1}{2.4} + \frac{1}{3.5} \dots + \frac{1}{n(n+2)} .$$