

§13. KIỂU BẢN GHI

Dữ liệu kiểu *bản ghi* (record) dùng để mô tả các đối tượng có cùng một số thuộc tính mà các thuộc tính có thể có các kiểu dữ liệu khác nhau.

Ví dụ, bảng kết quả thi gồm thông tin về các thí sinh như họ và tên, ngày sinh, giới tính, điểm các môn thi,... mà những thông tin này thuộc các kiểu dữ liệu khác nhau.

Bảng kết quả thi

Họ và tên	Ngày sinh	Giới tính	Điểm Tin	Điểm Toán	Điểm Lí	Điểm Hoá	Điểm Văn	Điểm Sử	Điểm Địa
Nguyễn Thị Minh Huệ	12/12/1990	Nữ	9	10	7	8	8	7	8
Dương Trúc Lâm	2/1/1990	Nam	9	10	8	8	9	6	7
Đào Văn Bình	5/12/1990	Nam	8	8	9	8	7	7	6
...

Một ví dụ khác, khi xem xét doanh số của một cửa hàng, ta quan tâm đến tập hoá đơn bán hàng, mỗi hoá đơn đều có các thuộc tính như tên hàng, đơn giá, chủng loại, số lượng bán, giá thành, người bán, người mua, ngày bán,...

Để mô tả các đối tượng như vậy, ngôn ngữ lập trình cho phép xác định kiểu dữ liệu bản ghi (trong C++ gọi là kiểu cấu trúc (struct)). Mỗi đối tượng được mô tả bằng một bản ghi. Mỗi thuộc tính của đối tượng tương ứng với một *trường* của bản ghi. *Các trường khác nhau có thể có các kiểu dữ liệu khác nhau.*

Ngôn ngữ lập trình đưa ra quy tắc, cách thức xác định:

- Tên kiểu bản ghi;
- Tên các thuộc tính (trường);
- Kiểu dữ liệu của mỗi trường;
- Cách khai báo biến;
- Cách tham chiếu đến trường.

Dưới đây giới thiệu cách khai báo kiểu, biến, tham chiếu đến trường và phép gán giá trị bản ghi trong Pascal.

1. Khai báo

Các thông tin cần khai báo bao gồm tên kiểu bản ghi, tên các thuộc tính, kiểu dữ liệu của mỗi thuộc tính.

Do dữ liệu kiểu bản ghi thường dùng để mô tả nhiều đối tượng nên ta thường định nghĩa một kiểu bản ghi và sau đó dùng nó để khai báo các biến liên quan.

Kiểu bản ghi thường được định nghĩa như sau:

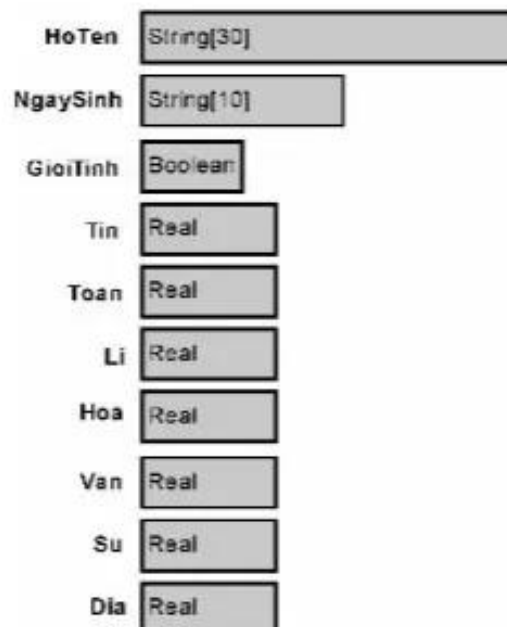
```
type <tên kiểu bản ghi> = record  
    <tên trường 1>: <kiểu trường 1>;  
    .....  
    <tên trường k>: <kiểu trường k>;  
end;
```

Sau khi có kiểu bản ghi, biến kiểu bản ghi có thể được khai báo như sau:

```
var  
    <tên biến bản ghi>: <tên kiểu bản ghi>;
```

Ví dụ

Để xử lý bảng kết quả thi nêu trên ta có thể khai báo *Lop* là biến mảng một chiều, mỗi phần tử mảng là một bản ghi *HocSinh* (dữ liệu về một học sinh). Mỗi bản ghi *HocSinh* gồm các thông tin: *HoTen*, *NgaySinh*, *GioiTinh* và điểm 7 môn thi: *Tin*, *Toan*, *Li*, *Hoa*, *Van*, *Su*, *Dia*.



Hình 14. Bản ghi *HocSinh*

Trong chương trình xử lý kết quả thi có thể sử dụng khai báo sau đây:

```
const Max =60; {gia thiet si so lop cao nhat la 60}
type
  HocSinh = record
    HoTen: string[30];
    NgaySinh: string[10];
    GioiTinh: boolean;
    Tin, Toan, Li, Hoa, Van, Su, Dia: Real;
  end;
var
  A, B: HocSinh;
  Lop: array[1..Max] of HocSinh;
```

Nếu A là biến kiểu bản ghi và X là tên một trường của A , thì để tham chiếu đến trường X , ta viết:

$A.X$

Để tham chiếu đến điểm tin học của một học sinh trong ví dụ trên ta viết:

$A.Tin$

2. Gán giá trị

Có hai cách để gán giá trị cho biến bản ghi:

- *Dùng lệnh gán trực tiếp:* Nếu A và B là hai biến bản ghi cùng kiểu, thì ta có thể gán giá trị của B cho A bằng câu lệnh:

$A := B;$

- *Gán giá trị cho từng trường:* Có thể thực hiện bằng lệnh gán hoặc nhập từ bàn phím.

Ví dụ, một lớp gồm N ($N \leq 60$) học sinh. Cần quản lý học sinh với các thuộc tính như họ và tên, ngày sinh, địa chỉ, điểm toán, điểm văn, xếp loại. Giả sử việc xếp loại được xác định như sau:

- Nếu tổng điểm toán và văn lớn hơn hoặc bằng 18 thì xếp loại A .
- Nếu tổng điểm toán và văn lớn hơn hoặc bằng 14 và nhỏ hơn 18 thì xếp loại B .

- Nếu tổng điểm toán và văn lớn hơn hoặc bằng 10 và nhỏ hơn 14 thì xếp loại *C*.
- Nếu tổng điểm toán và văn nhỏ hơn 10 thì xếp loại *D*.

Chú ý rằng, trong các thuộc tính cần quản lí, chỉ có năm thuộc tính đầu là độc lập, còn thuộc tính xếp loại được xác định dựa vào các điểm toán và văn. Để lưu trữ thông tin về học sinh, ta dùng kiểu bản ghi với sáu trường tương ứng với sáu thuộc tính cần quản lí.

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

```

program Xep_loai;
uses crt;
const Max = 60;
type HocSinh = record
    HoTen: string[30];
    NgaySinh: string[10];
    DiaChi: string[50];
    Toan, Van: real;
    XepLoai: char;
end;

var
    Lop: array[1..Max] of HocSinh;
    N, i: Byte;
begin
    clrscr;
    write('So luong hoc sinh trong lop N = '); readln(N);
    for i := 1 to N do
        begin
            writeln('Nhap so lieu ve hoc sinh thu ', i, ': ');
            write('Ho va ten: '); readln(Lop[i].HoTen);
            write('Ngay sinh: '); readln(Lop[i].NgaySinh);
            write('Dia chi: '); readln(Lop[i].DiaChi);
            write('Diem Toan: '); readln(Lop[i].Toan);
            write('Diem Van: '); readln(Lop[i].Van);
            if Lop[i].Toan+Lop[i].Van >= 18
                then Lop[i].XepLoai := 'A';
            if (Lop[i].Toan+Lop[i].Van >= 14) and
                (Lop[i].Toan+Lop[i].Van < 18)
                then Lop[i].XepLoai := 'B';
        end;
    end;

```

```

        if (Lop[i].Toan+Lop[i].Van>=10) and
            (Lop[i].Toan+Lop[i].Van<14)
            then Lop[i].XepLoai:='C';
        if (Lop[i].Toan+Lop[i].Van<10)
            then Lop[i].XepLoai:='D';
    end;
clrscr;
writeln('Danh sach xep loai hoc sinh trong lop:');
for i:= 1 to N do
    writeln(Lop[i].HoTen:30, ' - Xep loai: ',Lop[i].XepLoai);
readln
end.

```

TÓM TẮT

- Kiểu dữ liệu có cấu trúc được xây dựng từ những kiểu dữ liệu đã có theo quy tắc, khuôn dạng do ngôn ngữ lập trình cung cấp.
- Mảng một chiều
 - Mảng một chiều là dãy hữu hạn các phần tử cùng kiểu.
 - Khai báo: tên mảng, kiểu chỉ số, kiểu phần tử.
 - Tham chiếu phần tử mảng: *tên biến mảng[chỉ số phần tử]*
- Mảng hai chiều
 - Mảng hai chiều là bảng các phần tử cùng kiểu.
 - Khai báo: tên mảng, kiểu chỉ số hàng, kiểu chỉ số cột, kiểu phần tử.
 - Tham chiếu phần tử mảng: *tên biến mảng[chỉ số hàng, chỉ số cột]*
- Kiểu dữ liệu xâu
 - Xâu là dãy các kí tự trong bộ mã ASCII.
 - Các thao tác xử lí thường sử dụng:
 - Phép ghép xâu;
 - Phép so sánh;
 - Các thủ tục và hàm chuẩn xử lí xâu.
- Kiểu bản ghi
 - Khai báo: tên bản ghi, tên và kiểu các trường.
 - Tham chiếu trường của bản ghi: *tên biến bản ghi.tên trường*

CÂU HỎI VÀ BÀI TẬP

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

$$F_0 = 0; F_1 = 1; F_N = F_{N-1} + F_{N-2} \text{ với } N \geq 2.$$

Viết chương trình nhập từ bàn phím số nguyên dương N và đưa ra màn hình số hạng thứ N của dãy *Phi-bô-na-xi*. Chương trình của bạn thực hiện được với giá trị lớn nhất của N là bao nhiêu?

8. Chương trình sau đây thực hiện những gì?

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

```

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

```

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

Chương trình sau đây giải bài toán trên:

```

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

```

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

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