

Cấu trúc

Người soạn: Lương Thái Lê

Thời lượng: 3 tiết

Định nghĩa – Khai báo

- Cấu trúc là một kiểu dữ liệu do người dùng tự định nghĩa
- Định nghĩa một cấu trúc:

```
struct <Tên cấu trúc>
{
    <Kiểu 1> <Tên thành phần 1>;
    <Kiểu 2> <Tên thành phần 2>;
    ...
};
```

- Sau khi được định nghĩa, khai báo cấu trúc theo quy tắc:

```
<Tên cấu trúc> danh_sách_tên_biến_cấu_trúc;
```

Chú ý: <Tên cấu trúc> trở thành 1 kiểu dữ liệu chuẩn

Ví dụ

- Định nghĩa kiểu DiemThi:

```
struct DiemThi
{
    int Toan;
    int Ly;
    int Hoa;
};
```

- *Chú ý: có thể có tên (DiemThi) hoặc không → sự khác nhau.*
- Khai báo biến có kiểu DiemThi:

```
DiemThi diem1, diem2, diem3
```

Truy cập các thành phần của cấu trúc

- Toán tử này cho phép truy cập đến một thành phần cấu trúc khi biết tên biến cấu trúc và tên thành phần.
- Cú pháp:

`<biến_cấu_trúc> . <tên_thành_phần>`

- *Chú ý: Nếu thành phần của cấu trúc lại là 1 cấu trúc khác thì truy cập theo kiểu đệ quy*

`<biến_cấu_trúc> . <thành_phần_cấu_trúc> . <tên_thành_phần>`

Ví dụ về truy cập thành phần cấu trúc

```
Struct Cong_nhan
{
char ten[20];
char dia_chi[50];
float bac_luong;
Ngay ngay_sinh;
};
```

Khi đó có:

```
Cong_nhan.bac_luong = 3.5;
Cong_nhan.ngay_sinh.thang = 9;
```

Cách 1: có tên cấu trúc:

```
struct Ngay
{
    int ngay;
    int thang;
    int nam;
} ;
```

Ví dụ: Nhập dãy n điểm và in dãy điểm ra màn hình

```
#include<stdio.h>
main()
{
    struct DIEM
    {
        float x;
        float y;
    };
    int i,n;
    DIEM diem[20];
    printf("Nhap so diem n= ");scanf("%d",&n);
    for (i=1;i<=n;i++)
    {
        printf("x[%d] = ",i); scanf("%f",&diem[i].x);
        printf("y[%d] = ",i); scanf("%f",&diem[i].y);
    }
    printf("Day diem vua nhap la: \n");
    for (i=1;i<=n;i++)
        printf("(%.2f, %.2f)  ",diem[i].x,diem[i].y);
}
```

```

#include<stdio.h>
typedef struct
{
    float x;
    float y;
} DIEM;
void nhap_diem(DIEM diem[20], int n)
{
    int i;
    for (i=1;i<=n;i++)
    {
        printf("x[%d] = ",i); scanf("%f",&diem[i].x);
        printf("y[%d] = ",i); scanf("%f",&diem[i].y);
    }
}
void in_diem(DIEM diem[20],int n)
{
    int i;
    printf("Day diem vua nhap la: \n");
    for (i=1;i<=n;i++)
        printf("(%.2f, %.2f) ",diem[i].x,diem[i].y);
}
main()
{
    int n;
    DIEM diem[20];
    printf("Nhap so diem n= ");scanf("%d",&n);
    nhap_diem(diem,n);
    in_diem(diem,n);
}

```

```

#include<stdio.h>
typedef struct
{
    float x;
    float y;
} DIEM;
void nhap_diem(DIEM *diem, int n)
{
    int i;
    for (i=1;i<=n;i++)
    {
        printf("x[%d] = ",i); scanf("%f",&diem[i].x);
        printf("y[%d] = ",i); scanf("%f",&diem[i].y);
    }
}
void in_diem(DIEM *diem,int n)
{
    int i;
    printf("Day diem vua nhap la: \n");
    for (i=1;i<=n;i++)
        printf("(%.2f, %.2f) ",diem[i].x,diem[i].y);
}
main()
{
    int n;
    DIEM diem[20];
    printf("Nhap so diem n= ");scanf("%d",&n);
    nhap_diem(diem,n);
    in_diem(diem,n);
}

```

Viết hàm nhập, in, và tính độ dài đường gấp khúc của dãy điểm sử dụng cấu trúc

```
#include<stdio.h>
#include<math.h>
typedef struct
{
    float x;
    float y;
} DIEM;
void nhap_diem(DIEM *diem, int n)
{
    int i;
    for (i=1;i<=n;i++)
    {
        printf("x[%d] = ",i); scanf("%f",&diem[i].x);
        printf("y[%d] = ",i); scanf("%f",&diem[i].y);
    }
}
void in_diem(DIEM *diem, int n)
{
    int i;
    printf("Day diem vua nhap la: \n");
    for (i=1;i<=n;i++)
        printf("(%.2f, %.2f)  ",diem[i].x,diem[i].y);
}
```

```
float Do_dai_gap_khuc(DIEM *diem, int n)
{
    int i;
    float dx,dy,S=0;
    for (i=1;i<n;i++)
    {
        dx=(diem[i+1].x-diem[i].x);
        dy=(diem[i+1].y-diem[i].y);
        S+=sqrt(dx*dx+dy*dy);
    }
    return S;
}
main()
{
    int n;
    DIEM diem[20];
    printf("Nhap so diem n= ");scanf("%d",&n);
    nhap_diem(diem,n);
    in_diem(diem,n);
    printf("\nDo dai duong gap khuc la %.2f",Do_dai_gap_khuc(diem,n));
}
```