ĐƯỜNG TRÒN - CIRCLE

- Nguyễn Hoàng Yến Như
- Nguyễn Trần Phúc Nghi
- Nguyễn Trần Phúc An
- Nguyễn Đức Anh Phúc

- Trịnh Thị Thanh Trúc
- KS. Hồ Thái Ngọc
- KS. Cao Bá Kiệt
- KS. Lê Ngọc Huy
- CN. Bùi Cao Doanh
- CN. Nguyễn Trọng Thuận
- KS. Phan Vĩnh Long
- KS. Nguyễn Cường Phát
- ThS. Nguyễn Hoàng Ngân

- ThS. Đỗ Văn Tiến
- ThS. Nguyễn Hoàn Mỹ
- ThS. Dương Phi Long
- ThS. Trương Quốc Dũng
- ThS. Nguyễn Thành Hiệp
- ThS. Nguyễn Võ Đăng Khoa
- ThS. Võ Duy Nguyên
- ThS. Trần Việt Thu Phương
- TS. Nguyễn Tấn Trần Minh Khang



Khai báo kiểu dữ liệu

```
101.struct Diem
102.{
103.| float x;
104.| float y;
105.};
106.typedef struct Diem DIEM;
```



Khai báo kiểu dữ liệu

```
101.struct DuongTron
102.{
103.| DIEM I;
104.| float R;
105.};
106.typedef struct DuongTron DUONGTRON;
```



Nhập đường tròn



Nhập đường tròn

```
101.void <a href="https://www.nbap">Nhap</a>(<a href="https://www.nbap">DUONGTRON &c</a>)
102.{
103.
             cout << "Nhap tam: ";</pre>
             Nhap(c.I);
104.
             cout << "Nhap ban kinh:";</pre>
105.
             cin >> c.R;
106.
107.}
```



Xuất đường tròn



Xuất đường tròn



Tính chu vi

```
    – Định nghĩa hàm
```

```
101.float ChuVi(DUONGTRON c)
102.{
103.| return 2 * 3.14 * c.R;
104.}
```

$$P_c = 2\pi R$$



Tính diện tích

```
– Định nghĩa hàm
```

```
101.float DienTich(DUONGTRON c)
102.{
103.| return 3.14 * c.R * c.R;
104.}
```

$$S_c = \pi R^2$$



Kiểm tra một điểm thuộc đường tròn

```
101.float KhoangCach(DIEM P, DIEM Q)
102.{
103.
       return sqrt((P.x-Q.x)*(P.x-Q.x)
                   +(P.y-Q.y)*(P.y-Q.y));
104.
105.}
```



Kiểm tra một điểm thuộc đường tròn

```
    – Định nghĩa hàm

101.int <a href="https://ktthuoc(DUONGTRON c,DIEM P">KtThuoc(DUONGTRON c,DIEM P)</a>
102.{
103.
            float kc = KhoangCach(c.I, P);
104.
            if(kc<=c.R)</pre>
105.
                  return 1;
106.
            return 0;
107.}
```

Vị trí tương đối điểm và đường tròn



- Xét vị trí tương đối giữa một điểm và một đường tròn (nằm trong đường tròn, nằm trên đường tròn, nằm ngoài đường tròn).
- Giá trị trả về: Hàm trả về một trong các giá trị sau:
 - + 0. Điểm nằm trong đường tròn.
 - + 1. Điểm nằm trên đường tròn.
 - + 2. Điểm nằm ngoài đường tròn.
- Khai báo hàm
- 101.int TuongDoi(DUONGTRON,DIEM);

Vị trí tương đối của điểm và đường tròn

```
– Định nghĩa hàm
```

```
101.int TuongDoi(DUONGTRON c,DIEM P)
102.{
103.
          float kc = KhoangCach(c.I,P);
104.
          if(kc<c.R)</pre>
105.
               return 0;
106.
          if(kc==c.R)

    Giá trị trả về: Hàm trả về một trong các giá trị sau:

107.
               return 1;
                                + 0. Điểm nằm trong đường tròn.
108.
          return 2;
                                + 1. Điểm nằm trên đường tròn.
109.}
                                + 2. Điểm nằm ngoài đường tròn.
```

Vị trí tương đối đường thẳng và Circle

- Xét vị trí tương đối giữa một đường thắng và một đường tròn (không cắt, tiếp tuyến, cắt đường tròn).
- Giá trị trả về: Hàm trả về một trong các giá trị sau:
 - + 0. Đường thẳng không cắt đường tròn.
 - + 1. Đường thẳng tiếp xúc với đường tròn.
 - + 2. Đường thẳng cắt đường trong tại hai điểm.
- Khai báo hàm.
- 101.int TuongDoi(DUONGTRON, DUONGTHANG);

Vị trí tương đối đường thẳng và Circle



- Giá trị trả về: Hàm trả về một trong các giá trị sau:
 - + 0. Đường thẳng không cắt đường tròn.
 - + 1. Đường thẳng tiếp xúc với đường tròn.
 - + 2. Đường thẳng cắt đường trong tại hai điểm.

Vị trí tương đối đường thẳng và Circle



```
– Định nghĩa hàm
```

```
101.int TuongDoi(DUONGTRON c, DUONGTHANG d)
102.{
103.
          float kc = KhoangCach(c.I,d);
104.
          if(kc>c.R)
105.
               return 0;
106.
          if(kc==c.R)

    Giá trị trả về: Hàm trả về một trong các giá trị sau:

107.
               return 1;
                               + 0. Đường thẳng không cắt đường tròn.
108.
          return 2;
                               + 1. Đường thẳng tiếp xúc với đường tròn.
109.}
                               + 2. Đường thẳng cắt đường trong tại hai điểm.
```

Vị trí tương đối của hai đường tròn



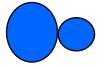
- Các trường hợp:
 - + Trùng nhau



+ Rời nhau



+ Tiếp xúc ngoài



+ Cắt nhau



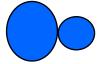
+ Tiếp xúc trong



- Hàm trả về một trong 6 giá trị
 - + 0. Trùng nhau



+ 2. Tiếp xúc ngoài



+ 3. Cắt nhau ()





+ Chứa trong nhau





Vị trí tương đối của hai đường tròn

```
— Định nghĩa hàm
101.float KhoangCach(DIEM P,DIEM Q)
```

```
102.{
103.| return sqrt((P.x-Q.x)*(P.x-Q.x)
104.| +(P.y-Q.y)*(P.y-Q.y));
105.}
```





```
101.int TuongDoi(DUONGTRON c1, DUONGTRON c2) — Hàm trả về một trong 6 giá trị
102.{
                                                    + 0. Trùng nhau
         float kc = KhoangCach(c1.I,c2.I);
103.
         if(kc==0 \&\& c1.R==c2.R)
104.
                                                    + 1. Rời nhau
105.
             return 0;
                                                    + 2. Tiếp xúc ngoài
         if(kc>c1.R+c2.R)
106.
107.
             return 1;
                                                    + 3. Cắt nhau
         if(kc==c1.R+c2.R)
108.
109.
             return 2;
                                                    + 4. Tiếp xúc trong
         if(kc<c1.R+c2.R && kc>abs(c1.R-c2.R)
110.
111.
             return 3;
                                                    + 5. Chứa trong nhau
         if(kc==abs(c1.R-c2.R))
112.
113.
             return 4;
114.
         return 5;
115.}
```



Chúc các bạn học tốt TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN TP.HCM

Nhóm UIT-Together Nguyễn Tấn Trần Minh Khang





Chúc các bạn học tốt TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN TP.HCM

Nhóm UIT-Together Nguyễn Tấn Trần Minh Khang