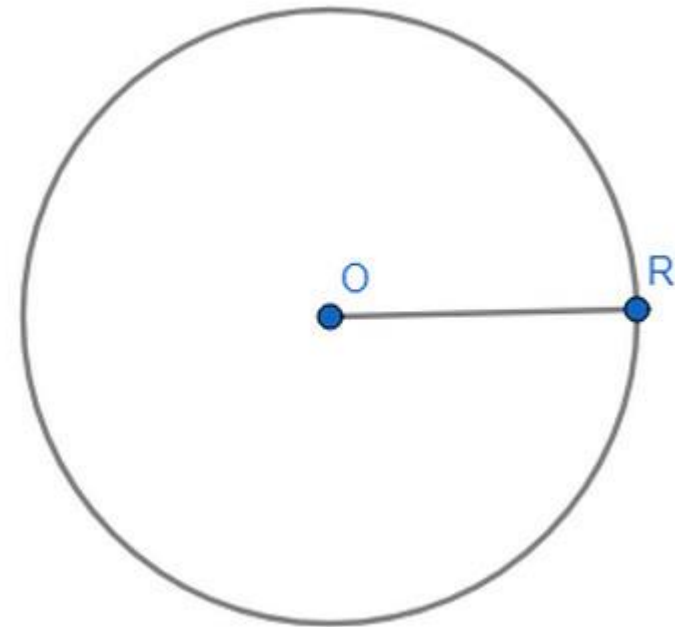


Chương 9 – THIẾT KẾ LỚP ĐƯỜNG TRÒN

- Nguyễn Hữu Lợi
- Đoàn Chánh Thống
- ThS. Nguyễn Thành Hiệp
- ThS. Trương Quốc Dũng
- ThS. Võ Duy Nguyên
- ThS. Nguyễn Văn Toàn
- TS. Nguyễn Duy Khánh
- TS. Nguyễn Tấn Trần Minh Khang

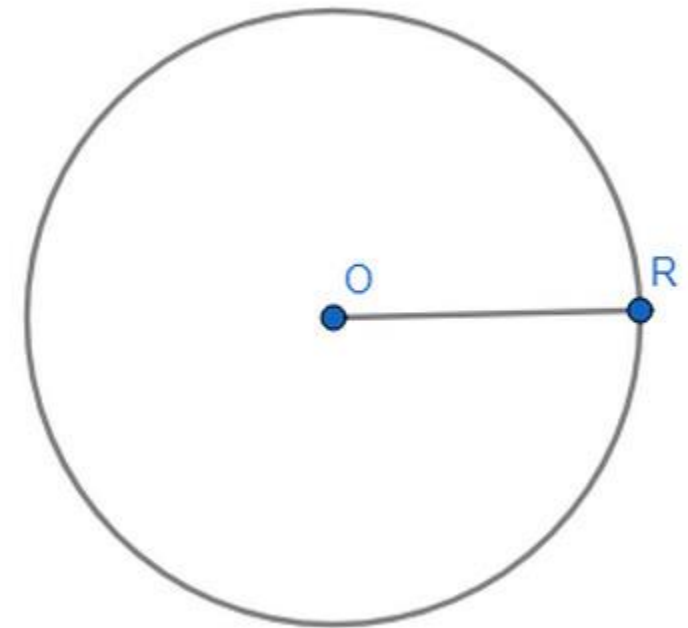
Thiết kế lớp CDuongTron

- Thuộc tính
 - + Tâm.
 - + Bán kính.



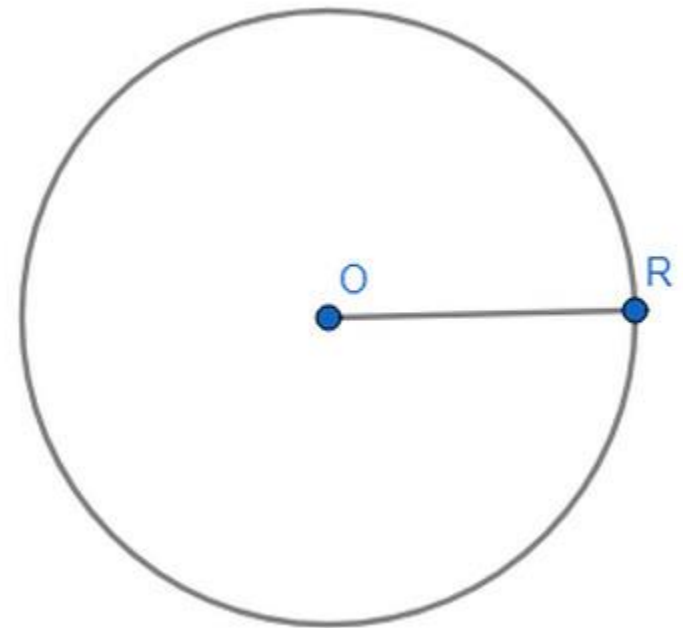
Thiết kế lớp CDuongTron

- Các nhóm phương thức
 - + Nhóm phương thức khởi tạo.
 - + Nhóm phương thức cung cấp thông tin.
 - + Nhóm phương thức cập nhật thông tin.
 - + Nhóm phương thức kiểm tra.
 - + Nhóm phương thức xử lý.



Thiết kế lớp CDuongTron

- Nhóm phương thức khởi tạo
 - + Phương thức Nhập.
 - + Toán tử vào (operator >>).
 - + ...



Thiết kế lớp CDuongTron

– Nhóm phương thức khởi tạo (tiếp tục)

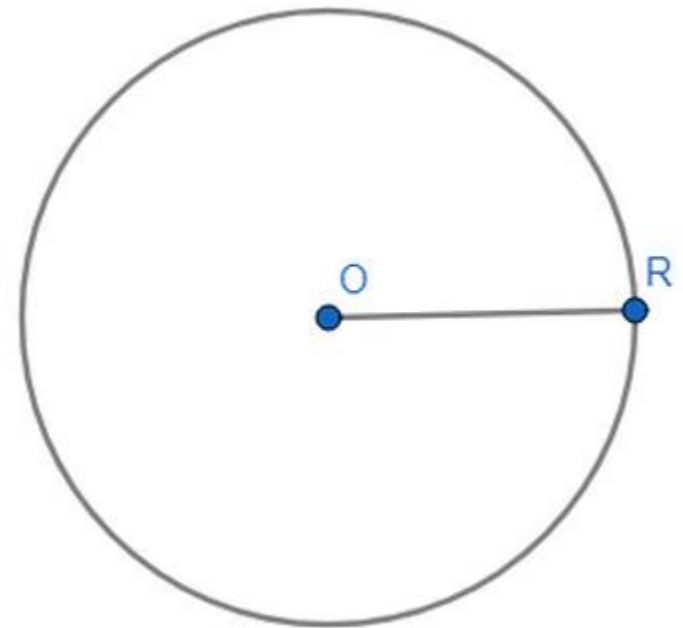
+ ...

+ Phương thức thiết lập mặc định.

+ Phương thức thiết lập sao chép.

+ Phương thức thiết lập khi biết đầy đủ thông tin.

+ ...



Thiết kế lớp CDuongTron

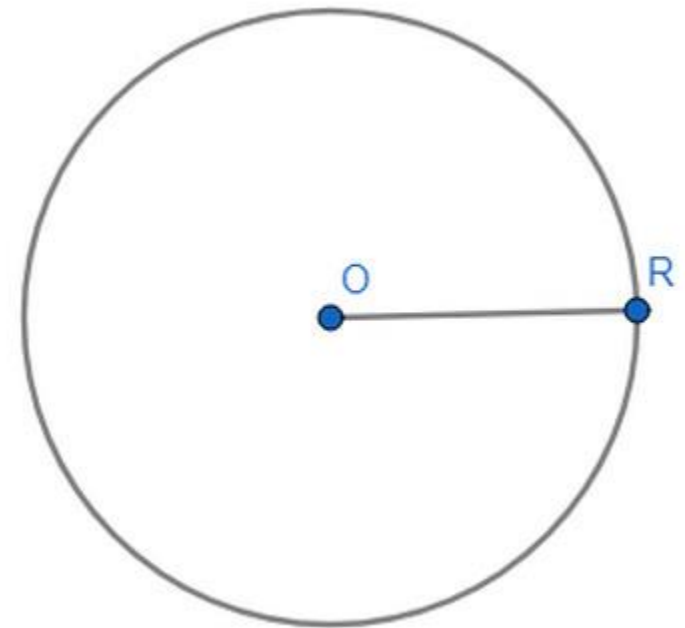
– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập khi biết tâm, bán kính lấy mặc định.

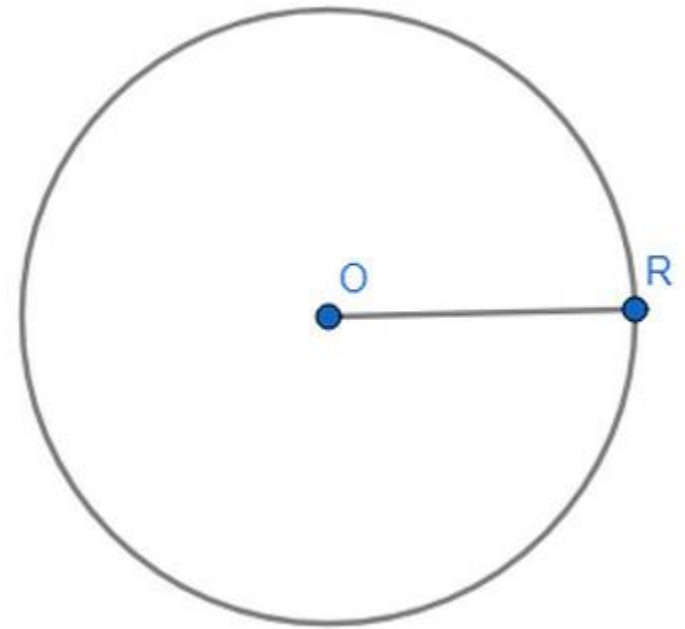
+ Phương thức thiết lập khi biết bán kính, tâm lấy mặc định.

+ ...



Thiết kế lớp CDuongTron

- Nhóm phương thức khởi tạo (tiếp tục)
 - + ...
 - + Phương thức khởi tạo mặc định.
 - + Phương thức khởi tạo sao chép.
 - + Phương thức khởi tạo khi biết đầy đủ thông tin.
 - + ...



Thiết kế lớp CDuongTron

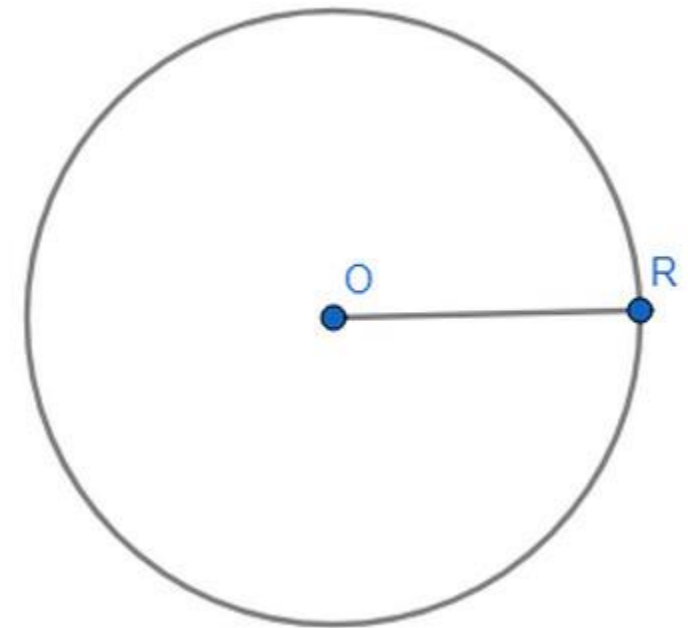
– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức khởi tạo khi biết tâm, bán kính lấy mặc định.

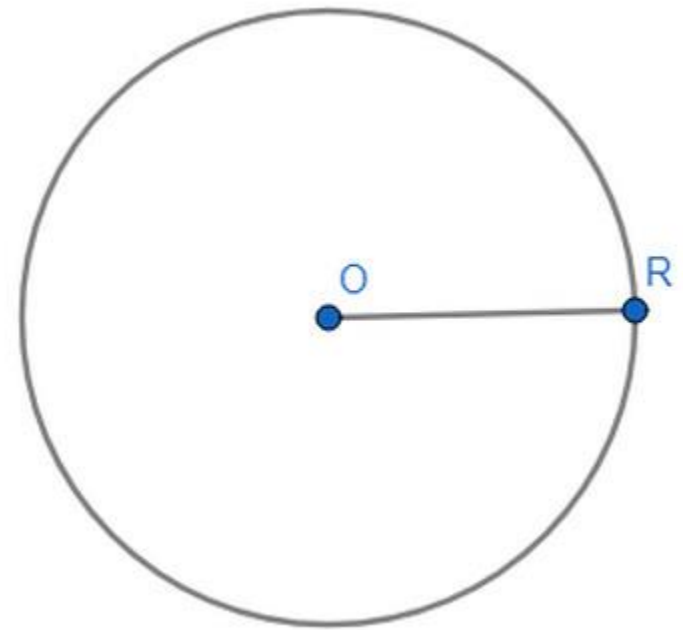
+ Phương thức khởi tạo khi biết bán kính, tâm lấy mặc định.

+ Kết thúc nhóm phương thức khởi tạo



Thiết kế lớp CDuongTron

- Nhóm phương thức cung cấp thông tin
 - + Phương thức Xuất.
 - + Toán tử ra (operator <<).
 - + ...
- + Lớp CDuongTron có bao nhiêu thuộc tính.
- + Trả lời: Lớp CDuongTron có 2 thuộc tính.



Thiết kế lớp CDuongTron

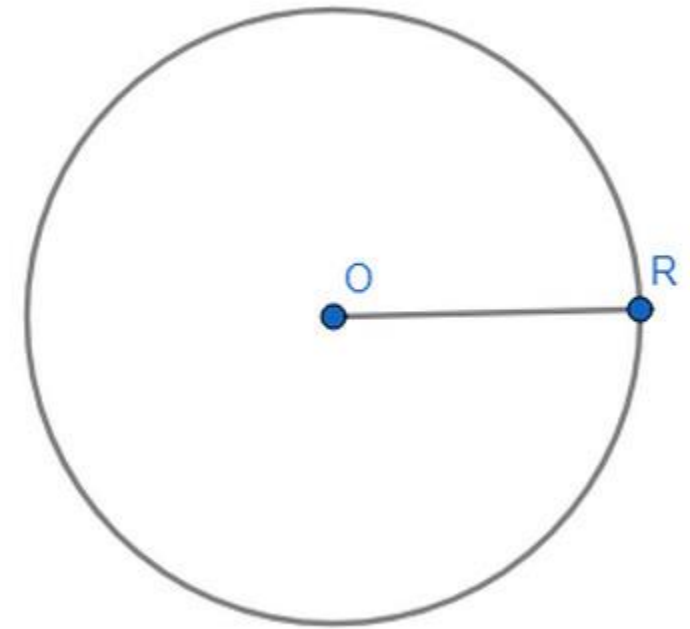
– Nhóm phương thức cung cấp thông tin (tiếp tục)

+ ...

+ Phương thức cung cấp tâm.

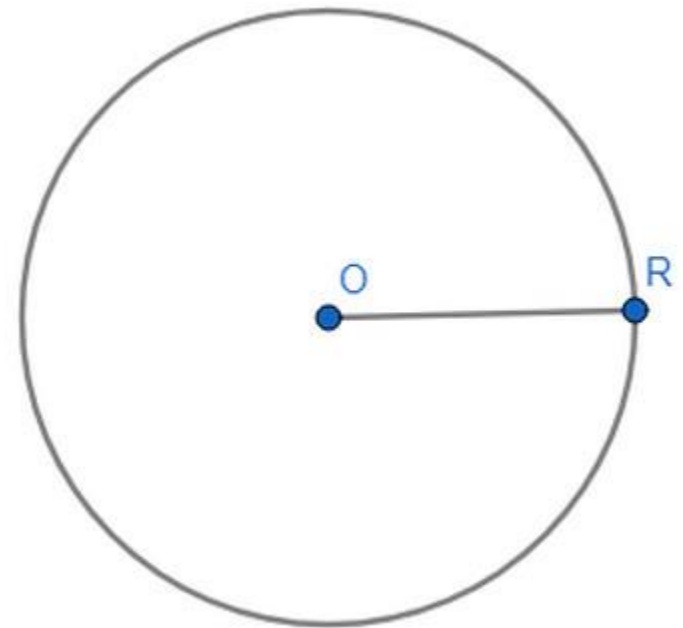
+ Phương thức cung cấp bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Thiết kế lớp CDuongTron

- Nhóm phương thức cập nhật thông tin
 - + Toán tử gán (operator =).
 - + ...
- + Lớp CDuongTron có bao nhiêu thuộc tính.
- + Trả lời: Lớp CDuongTron có 2 thuộc tính.



Thiết kế lớp CDuongTron

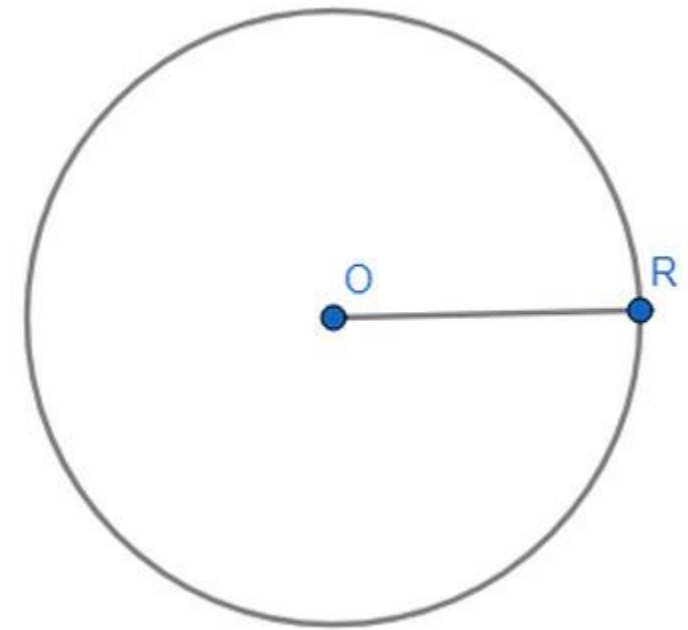
– Nhóm phương thức cập nhật thông tin (tiếp tục)

+ ...

+ Phương thức cập nhật tâm.

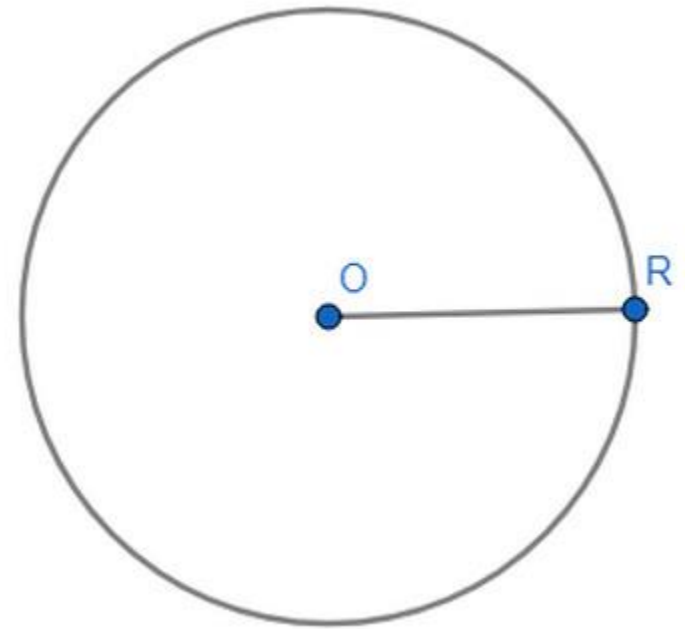
+ Phương thức cập nhật bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Thiết kế lớp CDuongTron

- Nhóm phương thức kiểm tra
 - + Toán tử so sánh bằng.
 - + Toán tử so sánh khác.
 - + Toán tử so sánh lớn hơn.
 - + Toán tử so sánh nhỏ hơn.
 - + Toán tử so sánh lớn hơn bằng.
 - + Toán tử so sánh nhỏ hơn bằng.
 - + ...



Thiết kế lớp CDuongTron

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra đường tròn có đi qua gốc tọa độ không?

+ Kiểm tra đường tròn có chứa gốc tọa độ không?

+ Kiểm tra đường tròn có tiếp xúc trục hoành không?

+ Kiểm tra đường tròn có tiếp xúc trục tung không?

+ Kiểm tra đường tròn là đường tròn đơn vị không?

+ ...

Thiết kế lớp CDuongTron

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra đường tròn có thuộc phần tư thứ 1 không?

+ Kiểm tra đường tròn có thuộc phần tư thứ 2 không?

+ Kiểm tra đường tròn có thuộc phần tư thứ 3 không?

+ Kiểm tra đường tròn có thuộc phần tư thứ 4 không?

+ ...

Thiết kế lớp CDuongTron

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra hai đường tròn có trùng nhau không?

+ Kiểm tra hai đường tròn có cắt nhau không?

+ Kiểm tra hai đường tròn có tiếp xúc trong không?

+ Kiểm tra hai đường tròn có tiếp xúc ngoài không?

+ Kiểm tra hai đường tròn có nằm trong nhau không?

+ Kiểm tra hai đường tròn có nằm ngoài nhau không?

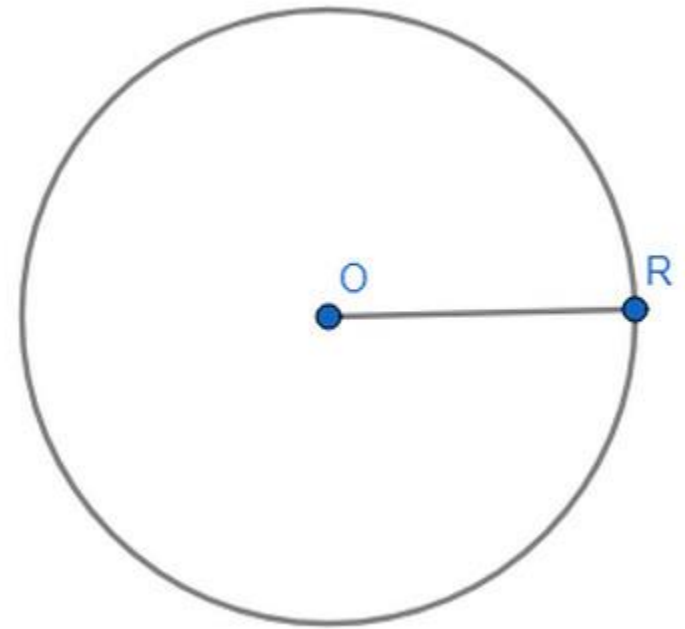
+ ...

Thiết kế lớp CDuongTron

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai đường tròn có đồng tâm không?
 - + ...
- + Kết thúc nhóm phương thức kiểm tra.

Thiết kế lớp CDuongTron

- Nhóm phương thức xử lý
 - + Phương thức phá hủy.
 - + ...



Thiết kế lớp CDuongTron

– Nhóm phương thức xử lý (tiếp tục)

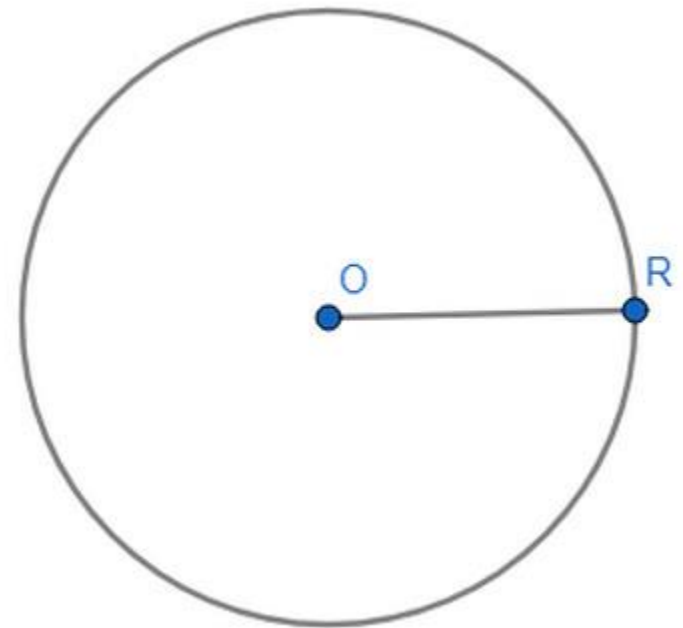
+ ...

+ Tính khoảng cách đường tròn đến gốc tọa độ.

+ Tính chu vi.

+ Tính diện tích.

+ ...



Thiết kế lớp CDuongTron

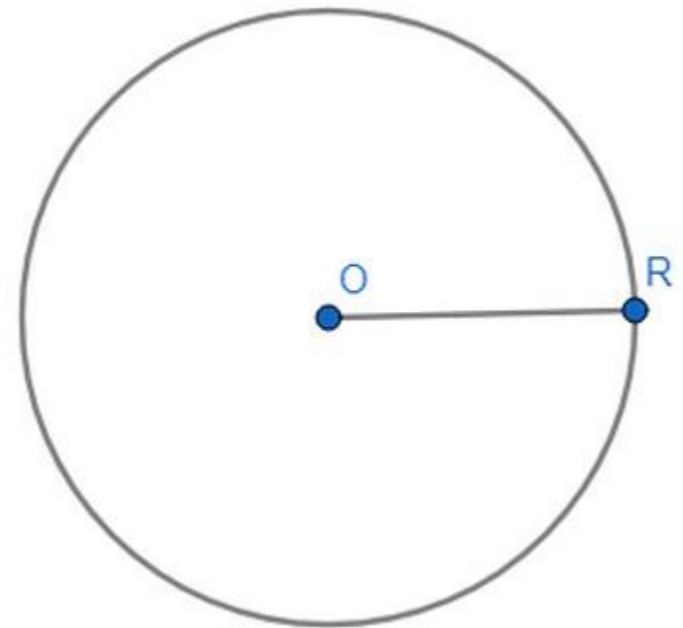
– Nhóm phương thức xử lý (tiếp tục)

+ ...

+ Tính diện tích phần giao của hai đường tròn.

+ Tính diện tích phần phủ của hai đường tròn.

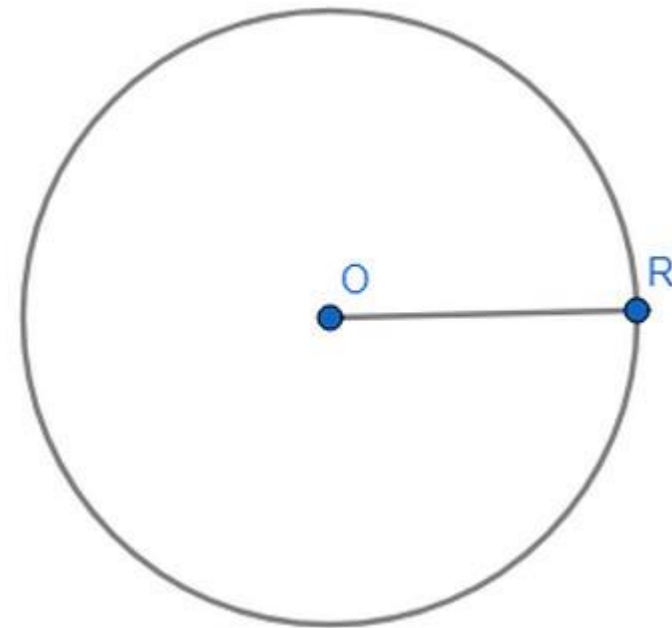
+ Kết thúc nhóm phương thức xử lý.



KHAI BÁO LỚP CDUONGTRON

Thiết kế lớp CDuongTron

- Thuộc tính
 - + Tâm.
 - + Bán kính.

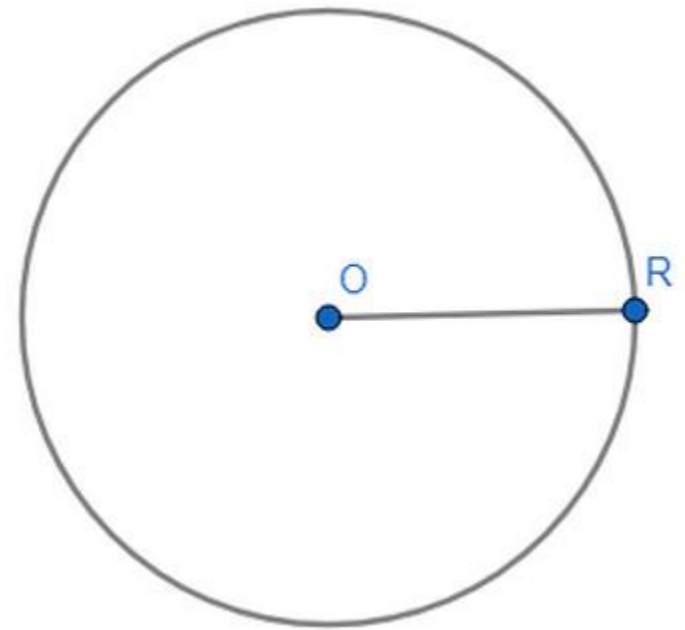


Khai báo lớp CDuongTron

```
11.class CDuongTron
12.{
13.    private:
14.        CDiem I;
15.        float R;
16.    public:
17.        ///// Nhóm phương thức khởi tạo
18.        ...
```

Thiết kế lớp CDuongTron

- Nhóm phương thức khởi tạo
 - + Phương thức Nhập.
 - + Toán tử vào (operator >>).
 - + ...



Khai báo lớp CDuongTron

```
11. | ...
12. | public:
13. |     ///// Nhóm phương thức khởi tạo
14. |     ///// Chương 03 - Lập Trình HĐT với C++
15. |     void Nhap();
16. |     ///// Chương 04 - Iostream Cơ Bản
17. |     friend istream& operator >>(istream&,
18. |                                   CDuongTron&);
19. |     ...
```

Thiết kế lớp CDuongTron

– Nhóm phương thức khởi tạo (tiếp tục)

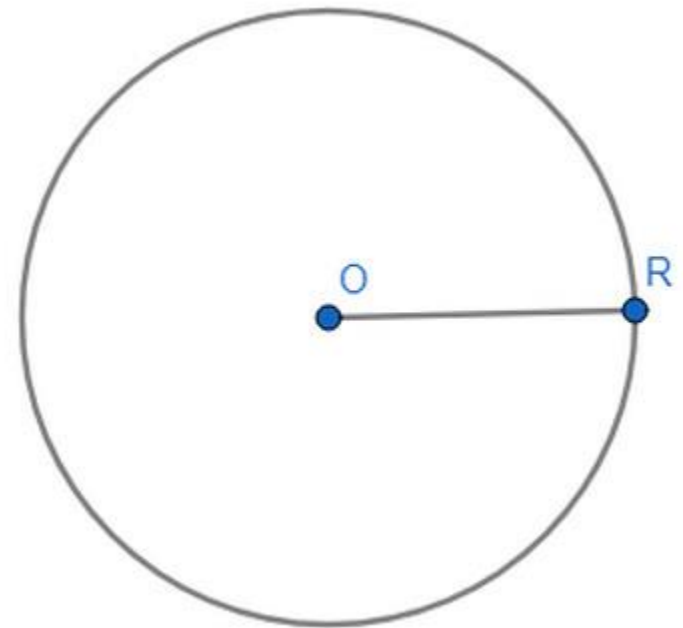
+ ...

+ Phương thức thiết lập mặc định.

+ Phương thức thiết lập sao chép.

+ Phương thức thiết lập khi biết đầy đủ thông tin.

+ ...



Thiết kế lớp CDuongTron

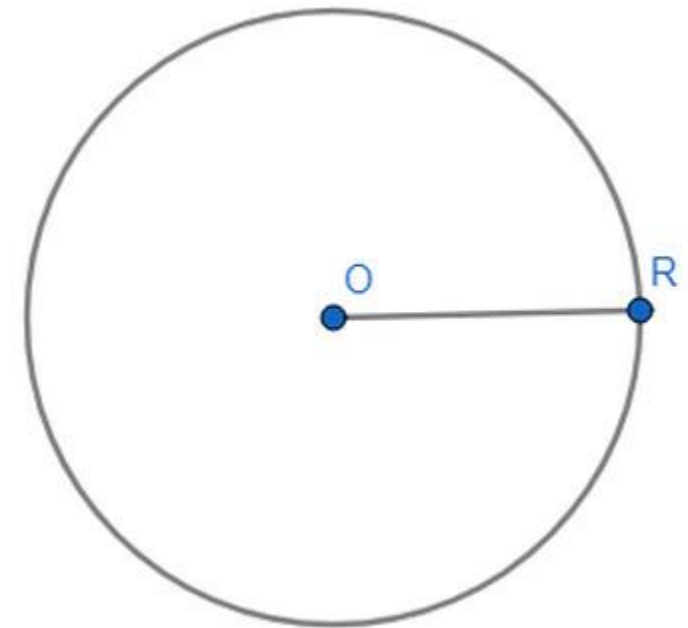
– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập khi biết tâm, bán kính lấy mặc định.

+ Phương thức thiết lập khi biết bán kính, tâm lấy mặc định.

+ ...

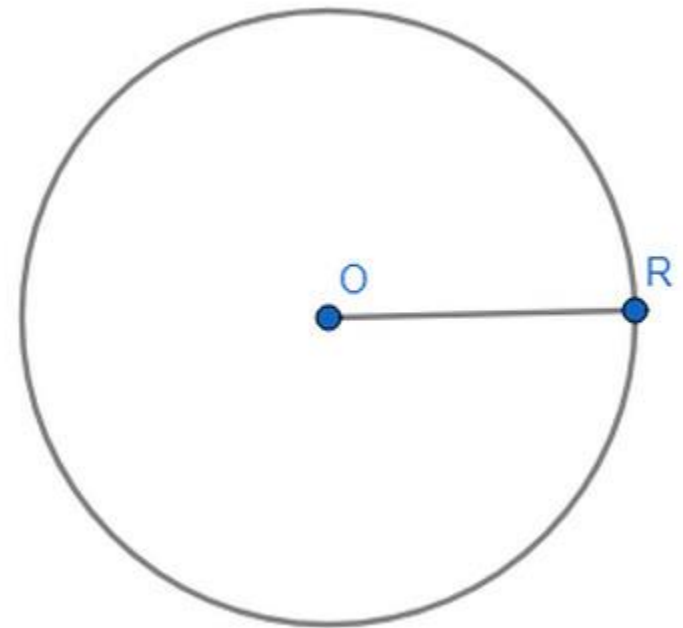


Khai báo lớp CDuongTron

```
11.  // /// Nhóm phương thức khởi tạo
12.  // /// Chương 05 - Phương thức thiết lập -
13.  // /// Phương thức phá hủy
14.  ...
15.  CDuongTron();
16.  CDuongTron(const CDuongTron&);
17.  CDuongTron(CDiem, float);
18.  CDuongTron(CDiem);
19.  CDuongTron(float);
```

Thiết kế lớp CDuongTron

- Nhóm phương thức khởi tạo (tiếp tục)
 - + ...
 - + Phương thức khởi tạo mặc định.
 - + Phương thức khởi tạo sao chép.
 - + Phương thức khởi tạo khi biết đầy đủ thông tin.
 - + ...



Thiết kế lớp CDuongTron

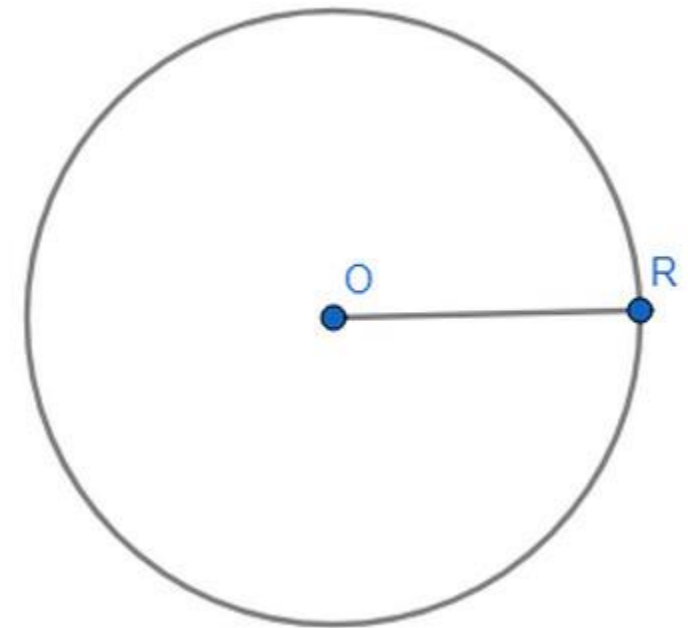
– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức khởi tạo khi biết tâm, bán kính lấy mặc định.

+ Phương thức khởi tạo khi biết bán kính, tâm lấy mặc định.

+ Kết thúc nhóm phương thức khởi tạo

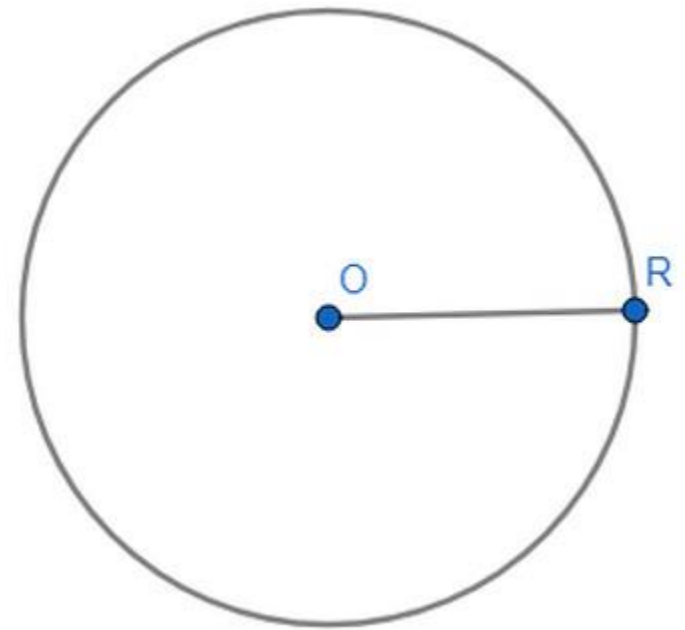


Khai báo lớp CDuongTron

```
11.  ///// Nhóm phương thức khởi tạo
12.  ///// Chương 09 - Thiết kế lớp
13.  ...
14.  void KhoiTao();
15.  void KhoiTao(const CDuongTron&);
16.  void KhoiTao(CDiem, float);
17.  void KhoiTao(CDiem);
18.  void KhoiTao(float);
19.  ///// Nhóm phương thức cung cấp thông tin
```

Thiết kế lớp CDuongTron

- Nhóm phương thức cung cấp thông tin
 - + Phương thức Xuất.
 - + Toán tử ra (operator <<).
 - + ...
- + Lớp CDuongTron có bao nhiêu thuộc tính.
- + Trả lời: Lớp CDuongTron có 2 thuộc tính.



Khai báo lớp CDuongTron

```
11.      ...
12.      ///// Nhóm phương thức cung cấp thông tin
13.      ///// Chương 03 - Lập Trình HĐT với C++
14.      void Xuat();
15.      ///// Chương 04 - Iostream Cơ Bản
16.      friend ostream& operator <<(ostream&,
17.                                     CDuongTron&);
18.      ...
```

Thiết kế lớp CDuongTron

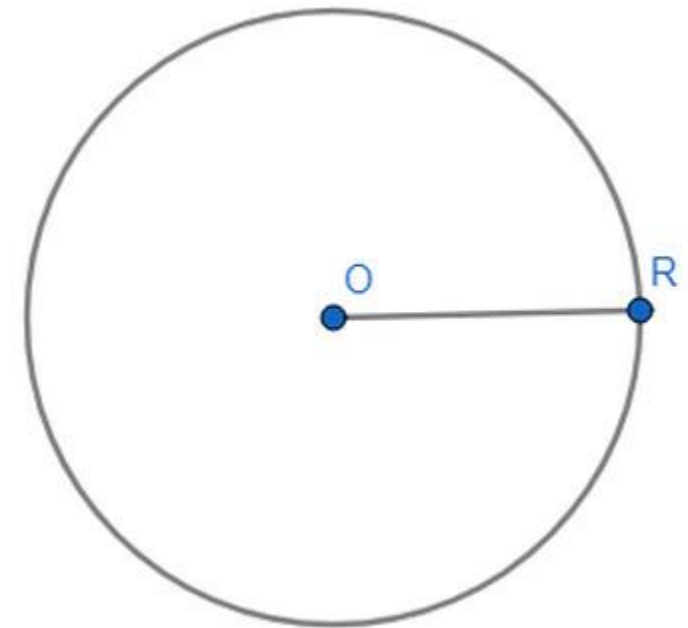
– Nhóm phương thức cung cấp thông tin (tiếp tục)

+ ...

+ Phương thức cung cấp tâm.

+ Phương thức cung cấp bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



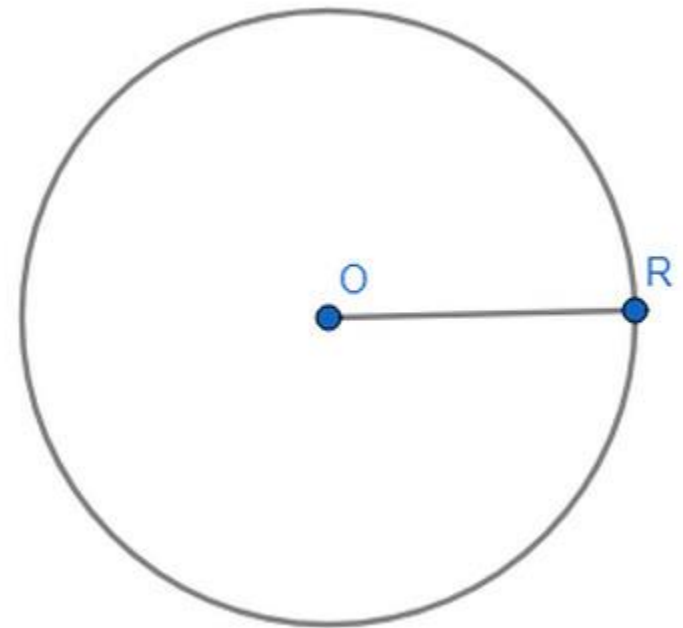
Khai báo lớp CDuongTron

```
11.      ///// Nhóm phương thức cung cấp thông tin
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      CDiem getI();
15.      float getR();

16.      ///// Nhóm phương thức cập nhật thông tin
17.      ...
```

Thiết kế lớp CDuongTron

- Nhóm phương thức cập nhật thông tin
 - + Toán tử gán (operator =).
 - + ...
- + Lớp CDuongTron có bao nhiêu thuộc tính.
- + Trả lời: Lớp CDuongTron có 2 thuộc tính.



Thiết kế lớp CDuongTron

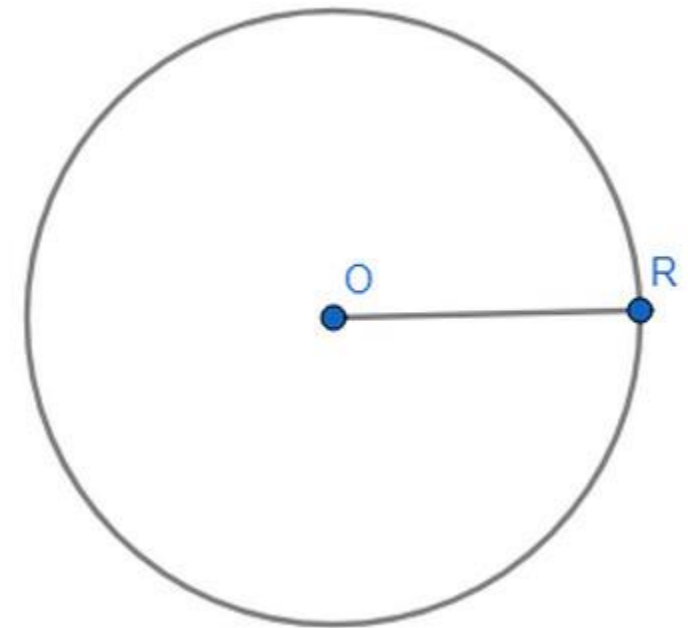
– Nhóm phương thức cập nhật thông tin (tiếp tục)

+ ...

+ Phương thức cập nhật tâm.

+ Phương thức cập nhật bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.

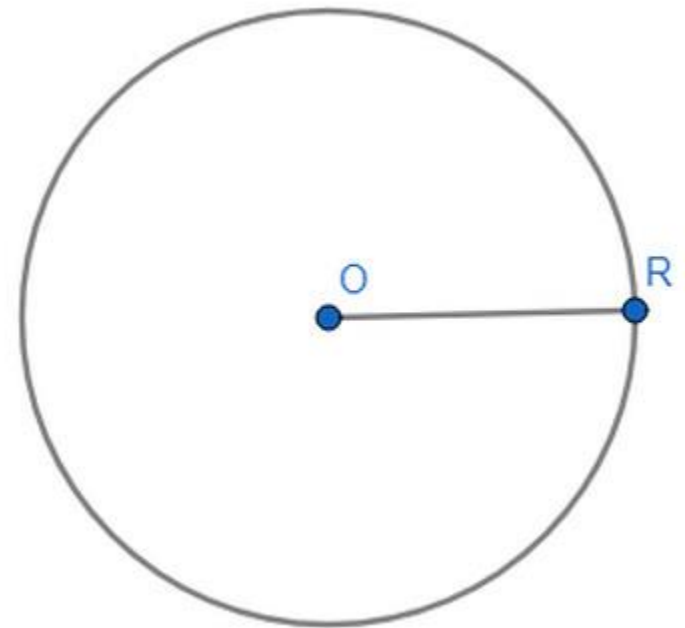


Khai báo lớp CDuongTron

```
11. ...  
12. ///// Nhóm phương thức cập nhật thông tin  
13. ///// Chương 06 - Toán tử gán  
14. CDuongTron& operator =(const CDuongTron&);  
15. ///// Chương 09 - Thiết kế lớp  
16. void setI(CDiem);  
17. void setR(float);  
18. ///// Nhóm phương thức kiểm tra  
19. ...
```

Thiết kế lớp CDuongTron

- Nhóm phương thức kiểm tra
 - + Toán tử so sánh bằng.
 - + Toán tử so sánh khác.
 - + Toán tử so sánh lớn hơn.
 - + Toán tử so sánh nhỏ hơn.
 - + Toán tử so sánh lớn hơn bằng.
 - + Toán tử so sánh nhỏ hơn bằng.
 - + ...



Khai báo lớp CDuongTron

```
11. ...  
12. ///// Nhóm phương thức kiểm tra  
13. ///// Chương 08 - Toán tử so sánh  
14. bool operator==(const CDuongTron&);  
15. bool operator!=(const CDuongTron&);  
16. bool operator>(const CDuongTron&);  
17. bool operator<(const CDuongTron&);  
18. bool operator>=(const CDuongTron&);  
19. bool operator<=(const CDuongTron&);
```

Thiết kế lớp CDuongTron

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra đường tròn có đi qua gốc tọa độ không?

+ Kiểm tra đường tròn có chứa gốc tọa độ không?

+ Kiểm tra đường tròn có tiếp xúc trục hoành không?

+ Kiểm tra đường tròn có tiếp xúc trục tung không?

+ Kiểm tra đường tròn là đường tròn đơn vị không?

+ ...

Khai báo lớp CDuongTron

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      bool ktQuaGoc();
15.      bool ktChuaGoc();
16.      bool ktTiepXucHoanh();
17.      bool ktTiepXucTung();
18.      bool ktDonVi();
19.      ...
```

Thiết kế lớp CDuongTron

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra đường tròn có thuộc phần tư thứ 1 không?

+ Kiểm tra đường tròn có thuộc phần tư thứ 2 không?

+ Kiểm tra đường tròn có thuộc phần tư thứ 3 không?

+ Kiểm tra đường tròn có thuộc phần tư thứ 4 không?

+ ...

Khai báo lớp CDuongTron

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      bool ktPhanTu1();
15.      bool ktPhanTu2();
16.      bool ktPhanTu3();
17.      bool ktPhanTu4();
18.      ...
```

Thiết kế lớp CDuongTron

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra hai đường tròn có trùng nhau không?

+ Kiểm tra hai đường tròn có cắt nhau không?

+ Kiểm tra hai đường tròn có tiếp xúc trong không?

+ Kiểm tra hai đường tròn có tiếp xúc ngoài không?

+ Kiểm tra hai đường tròn có nằm trong nhau không?

+ Kiểm tra hai đường tròn có nằm ngoài nhau không?

+ ...

Khai báo lớp CDuongTron

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      bool ktTrung(const CDuongTron&);
15.      bool ktCat(const CDuongTron&);
16.      bool ktTiepXucTrong(const CDuongTron&);
17.      bool ktTiepXucNgoai(const CDuongTron&);
18.      bool ktNamTrong(const CDuongTron&);
19.      bool ktNamNgoai(const CDuongTron&);
```

Thiết kế lớp CDuongTron

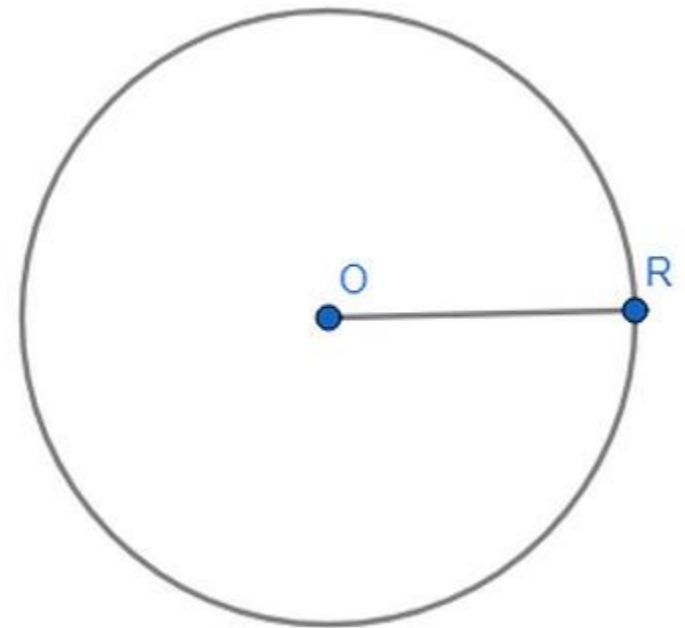
- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai đường tròn có đồng tâm không?
 - + ...
- + Kết thúc nhóm phương thức kiểm tra.

Khai báo lớp CDuongTron

```
11. |      ///// Nhóm phương thức kiểm tra
12. |      ///// Chương 09 - Thiết kế lớp
13. |      ...
14. |      bool ktDongTam(const CDuongTron&);
15. |      ///// Nhóm phương thức xu ly
```

Thiết kế lớp CDuongTron

- Nhóm phương thức xử lý
 - + Phương thức phá hủy.
 - + ...



Khai báo lớp CDuongTron

```
11. | ...  
12. | //// Nhóm phương thức xử lý  
13. | //// Chương 05 - Phương thức thiết lập -  
14. | //// Phương thức phá hủy  
15. | ~CDuongTron();  
16. | ...
```


Thiết kế lớp CDuongTron

– Nhóm phương thức xử lý (tiếp tục)

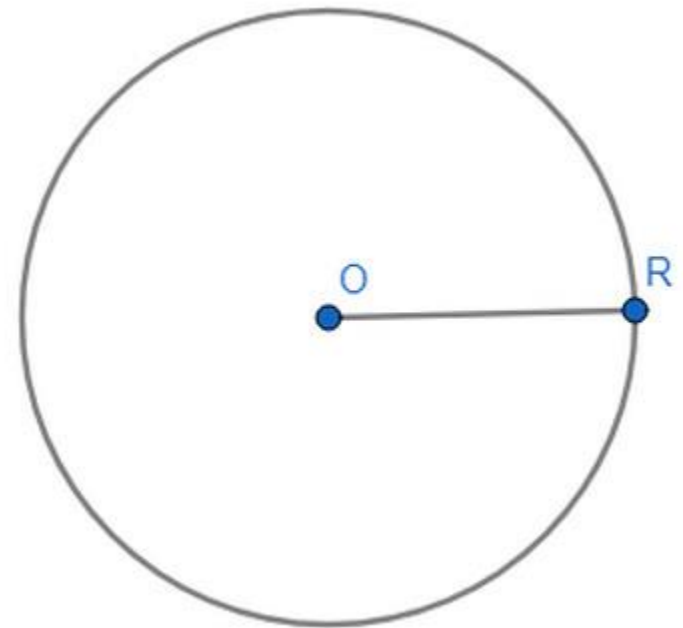
+ ...

+ Tính khoảng cách đường tròn đến gốc tọa độ.

+ Tính chu vi.

+ Tính diện tích.

+ ...



Khai báo lớp CDuongTron

```
11.  // /// Nhóm phương thức xử lý
12.  // /// Chương 09 - Thiết kế lớp
13.  ...
14.  float KhoangCachGoc();
15.  float ChuVi();
16.  float DienTich();
17.  ...
```

Thiết kế lớp CDuongTron

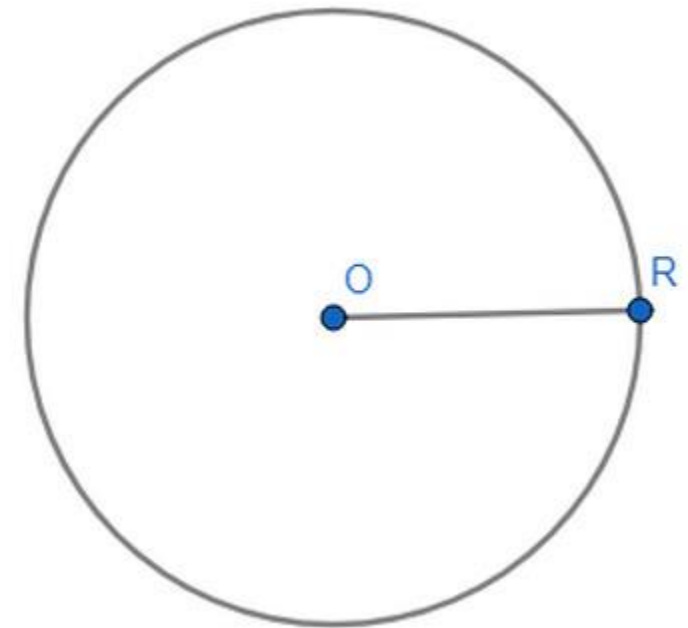
– Nhóm phương thức xử lý (tiếp tục)

+ ...

+ Tính diện tích phần giao của hai đường tròn.

+ Tính diện tích phần phủ của hai đường tròn.

+ Kết thúc nhóm phương thức xử lý.



Khai báo lớp CDuongTron

```
11.      //... Nhóm phương thức xử lý
12.      //... Chương 09 - Thiết kế lớp
13.      ...
14.      float DienTichPhu(const CDuongTron&);
15.      float DienTichGiao(const CDuongTron&);
16.};
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC KHỞI TẠO

Định nghĩa phương thức

```
11. void CDuongTron::Nhap()  
12. {  
13.     cout << "\nNhap toa do tam I: " << endl;  
14.     I.Nhap();  
15.     cout << "Nhap ban kinh R: ";  
16.     cin >> R;  
17. }
```

Định nghĩa phương thức

```
11.istream& operator >>(istream& is, CDuongTron& x)
12.{
13.    cout << "\nNhap I:";
14.    x.I.Nhap();
15.    cout << "Nhap R:";
16.    is >> x.R;
17.    return is;
18.}
```


Định nghĩa phương thức

```
11. CDuongTron::CDuongTron()
```

```
12. {
```

```
13. |   I = CDiem();
```

```
14. |   R = 0;
```

```
15. }
```

Định nghĩa phương thức

```
11. CDuongTron::CDuongTron(const CDuongTron& C)
```

```
12. {
```

```
13. |   I = C.I;
```

```
14. |   R = C.R;
```

```
15. }
```

Định nghĩa phương thức

```
11. CDuongTron::CDuongTron(CDiem II, float RR)
```

```
12. {
```

```
13. |   I = II;
```

```
14. |   R = RR;
```

```
15. }
```

Định nghĩa phương thức

```
11. CDuongTron::CDuongTron(CDiem II)
```

```
12. {
```

```
13. |   I = II;
```

```
14. |   R = 1;
```

```
15. }
```

Định nghĩa phương thức

```
11. CDuongTron::CDuongTron(float RR)
```

```
12. {
```

```
13. |   I = CDiem();
```

```
14. |   R = RR;
```

```
15. }
```

Định nghĩa phương thức

```
11. void CDuongTron::KhoiTao()  
12. {  
13. |   I = CDiem();  
14. |   R = 0;  
15. }
```

Định nghĩa phương thức

```
11. void CDuongTron::KhoiTao(const CDuongTron& C)
12. {
13. |   I = C.I;
14. |   R = C.R;
15. }
```


Định nghĩa phương thức

```
11. void CDuongTron::KhoiTao(CDiem II, float RR)
12. {
13. |   I = II;
14. |   R = RR;
15. }
```

Định nghĩa phương thức

```
11. void CDuongTron::KhoiTao(CDiem II)
12. {
13. |   I = II;
14. |   R = 1;
15. }
```

Định nghĩa phương thức

```
11. void CDuongTron::KhoiTao(float RR)
12. {
13. |   R = RR;
14. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC CUNG CẤP THÔNG TIN

Định nghĩa phương thức

```
11. void CDuongTron::Xuat()  
12. {  
13.     cout << "\n Toa do tam I: ";  
14.     I.Xuat();  
15.     cout << "\n Ban kinh R: ";  
16.     cout << R;  
17. }
```

Định nghĩa phương thức

```
11. ostream& operator <<(ostream& os, CDuongTron& x)
12. {
13.     cout << "\n Toa do tam I: ";
14.     x.I.Xuat();
15.     cout << "\n Ban kinh R: ";
16.     os << x.R;
17.     return os;
18. }
```

Định nghĩa phương thức

```
11. CDiem CDuongTron::getI()  
12. {  
13. |   return I;  
14. }
```


Định nghĩa phương thức

```
11.float CDuongTron::getR()  
12.{  
13.|    return R;  
14.}
```

Định nghĩa các phương thức
**NHÓM PHƯƠNG THỨC
CẬP NHẬT THÔNG TIN**

Định nghĩa phương thức

```
11.CDuongTron& CDuongTron::operator=(const CDuongTron&  
    C)  
12.{  
13.    I = C.I;  
14.    R = C.R;  
15.    return *this;  
16.}
```

Định nghĩa phương thức

```
11. void CDuongTron::setI(CDiem II)
12. {
13. |   I = II;
14. }
```

Định nghĩa phương thức

```
11. void CDuongTron::setR(float RR)
12. {
13. |   R = RR;
14. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC KIỂM TRA

Định nghĩa phương thức

```
11. bool CDuongTron::operator ==(const CDuongTron& c)
12. {
13.     if (R == c.R)
14.         return true;
15.     return false;
16. }
```


Định nghĩa phương thức

```
11. bool CDuongTron::operator!=(const CDuongTron& c)
12. {
13.     if (R != c.R)
14.         return true;
15.     return false;
16. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::operator>(const CDuongTron& c)
12. {
13.     if (R > c.R)
14.         return true;
15.     return false;
16. }
```

Định nghĩa phương thức

```
11.bool CDuongTron::operator<(const CDuongTron& c)
12.{
13.    if (R < c.R)
14.        return true;
15.    return false;
16.}
```

Định nghĩa phương thức

```
11. bool CDuongTron::operator>=(const CDuongTron& c)
12. {
13.     if (R >= c.R)
14.         return true;
15.     return false;
16. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::operator<=(const CDuongTron& c)
12. {
13.     if (R <= c.R)
14.         return true;
15.     return false;
16. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktQuaGoc()  
12. {  
13.     if (I.KhoangCachGoc() == R)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktChuaGoc()  
12. {  
13.     if (I.KhoangCachGoc() <= R)  
14.         return true;  
15.     return false;  
16. }
```


Định nghĩa phương thức

```
11.bool CDuongTron::ktTiepXucHoanh()  
12.{  
13.    if (R == abs(I.getX()))  
14.        return true;  
15.    return false;  
16.}
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktTiepXucTung()  
12. {  
13.     if (R == abs(I.getY()))  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11.bool CDuongTron::ktPhanTu1()  
12.{  
13.    if (R <= abs(I.getX()) && R <= abs(I.getY()))  
14.        if(I.ktThuoc1()==true)  
15.            return true;  
16.    return false;  
17.}
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktPhanTu2()  
12. {  
13.     if (R <= abs(I.getX()) && R <= abs(I.getY()))  
14.         if (I.ktThuoc2() == true)  
15.             return true;  
16.     return false;  
17. }
```

Định nghĩa phương thức

```
11.bool CDuongTron::ktPhanTu3()  
12.{  
13.    if (R <= abs(I.getX()) && R <= abs(I.getY()))  
14.        if (I.ktThuoc3() == true)  
15.            return true;  
16.    return false;  
17.}
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktPhanTu4()  
12. {  
13.     if (R <= abs(I.getX()) && R <= abs(I.getY()))  
14.         if (I.ktThuoc4() == true)  
15.             return true;  
16.     return false;  
17. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktDonVi()  
12. {  
13.     CDiem goc;  
14.     CDuongTron temp(goc, 1);  
15.     if (temp.ktTrung(*this))  
16.         return true;  
17.     return false;  
18. }
```


Định nghĩa phương thức

```
11. bool CDuongTron::ktTrung(CDuongTron c)
12. {
13.     if (c.I == I && c.R == R)
14.         return true;
15.     return false;
16. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktCat(const CDuongTron& c)
12. {
13.     float kc = I.KhoangCach(c.I);
14.     if ((R - c.R) < kc && kc < (R + c.R))
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktTiepXucTrong(const CDuongTron& c)
12. {
13.     float kc = I.KhoangCach(c.I);
14.     if (abs(R - c.R) == kc)
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktTiepXucNgoai(const CDuongTron& c)
12. {
13.     float kc = I.KhoangCach(c.I);
14.     if (abs(R + c.R) == kc)
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktNamTrong(const CDuongTron& c)
12. {
13.     float kc = I.KhoangCach(c.I);
14.     if (abs(R - c.R) < kc)
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktNamNgoai(const CDuongTron& c)
12. {
13.     float kc = I.KhoangCach(c.I);
14.     if (abs(aR + c.R) < kc)
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDuongTron::ktDongTam(const CDuongTron& c)
12. {
13.     if (I == c.I)
14.         return true;
15.     return false;
16. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC XỬ LÝ

Định nghĩa phương thức

```
11. CDuongTron::~CDuongTron()  
12. {  
13. |   return;  
14. }
```

Định nghĩa phương thức

```
11.float CDuongTron::KhoangCachGoc()  
12.{  
13.|   return abs(I.KhoangCachGoc() - R);  
14.}
```

Định nghĩa phương thức

```
11.float CDuongTron::ChuVi()  
12.{  
13.|    return (float) (2 * M_PI * R);  
14.}
```

Định nghĩa phương thức

```
11.float CDuongTron::DienTich()  
12.{  
13.|    return (float)(R * R * M_PI);  
14.}
```

Định nghĩa phương thức

```
11.float CDuongTron::DienTichPhu(const CDuongTron& c)
12.{
13.    float S1 = pow(R, 2) * M_PI;
14.    float S2 = pow(c.R, 2) * M_PI;
15.    return S1 + S2 - DienTichGiao(c);
16.}
```

Định nghĩa phương thức

```
11.float CDuongTron::DienTichGiao(const CDuongTron& c)
12.{
13.    float S1 = pow(R, 2) * M_PI;
14.    float S2 = pow(c.R, 2) * M_PI;
15.    float d = I.KhoangCach(c.I);

16.    float cos1 = (float)(pow(R, 2) + pow(d, 2) -
        pow(c.R, 2)) / (2 * R * d);
17.    float S11 = (float)(acos(cos1) * S1) / M_PI;
```

Cảm ơn quý vị đã lắng nghe

Nhóm tác giả

TS. Nguyễn Tấn Trần Minh Khang