

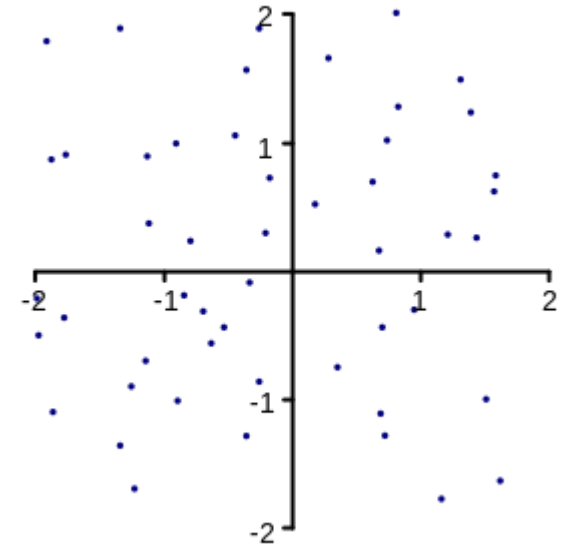
Chương 9 – THIẾT KẾ LỚP ĐIỂM TRONG MẶT PHẪNG OXY

- 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 ĐIỂM

Thiết kế lớp CDiem

- Thuộc tính
 - + Hoành độ.
 - + Tung độ.



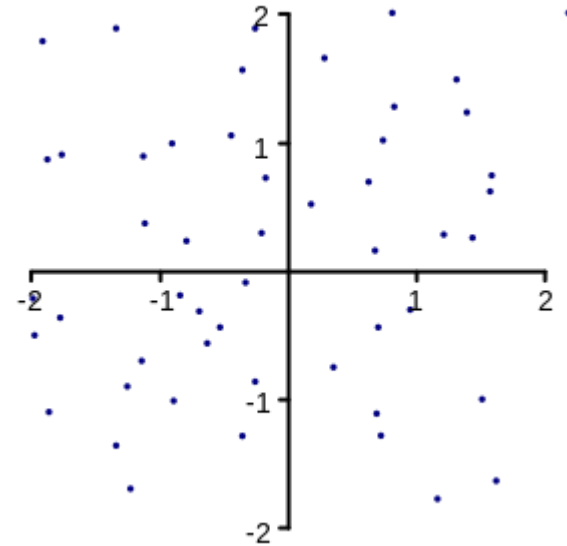
Thiết kế lớp CDiem

- 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 CDiem

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



Thiết kế lớp CDiem

– Nhóm phương thức khởi tạo (tiếp tục)

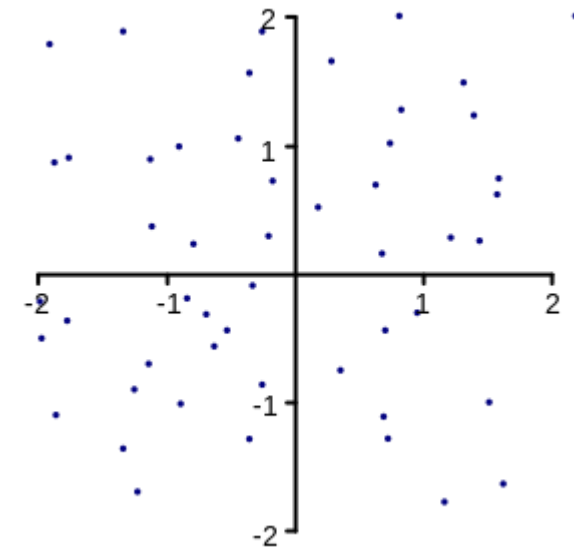
+ ...

+ Phương thức thiết lập mặc định.

+ Phương thức thiết lập sao chép.

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

+ ...



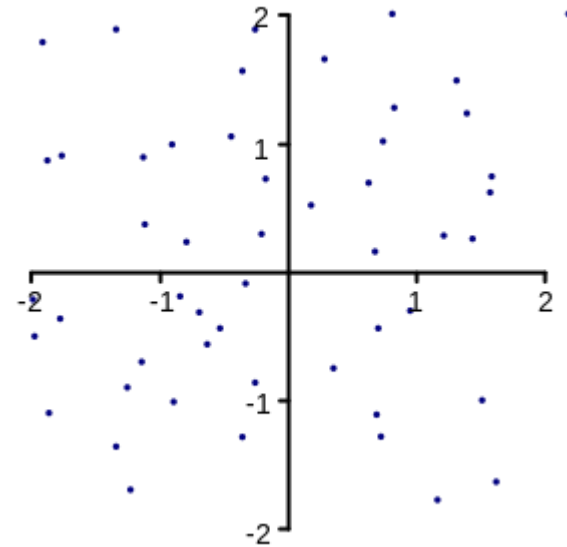
Thiết kế lớp CDim

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

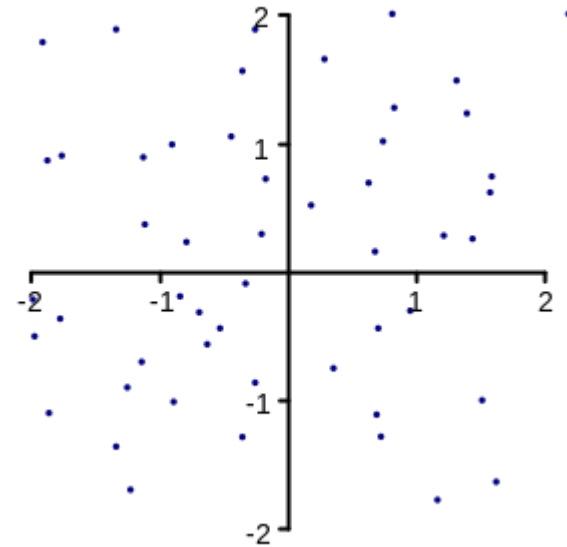
+ Phương thức thiết lập khi biết hoành độ, tung độ lấy mặc định.

+ ...



Thiết kế lớp CDiem

- 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.
 - + ...

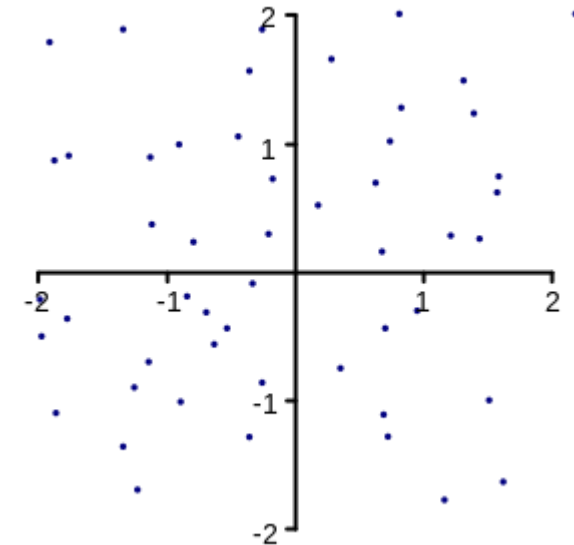


Thiết kế lớp CDim

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

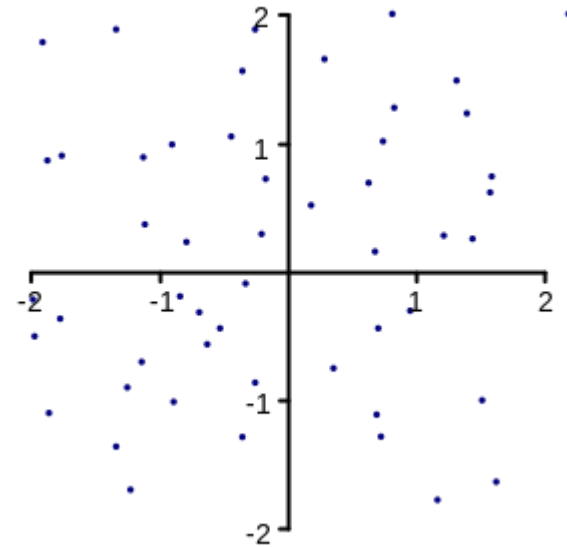
+ Phương thức khởi tạo khi biết hoành độ, tung độ lấy mặc định.



+ Kết thúc nhóm phương thức khởi tạo

Thiết kế lớp CDiem

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



Thiết kế lớp CDiem

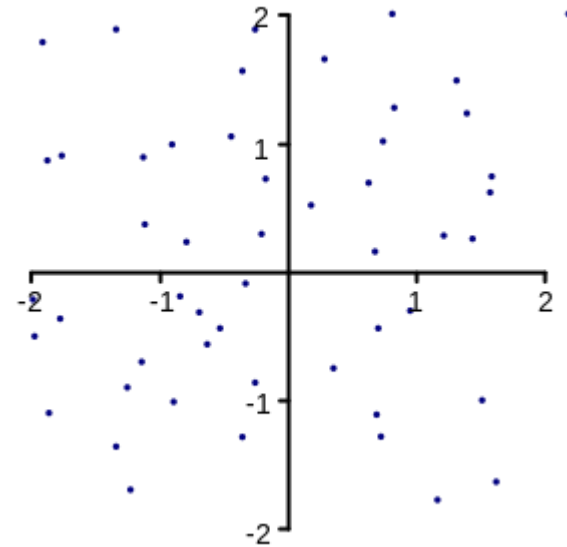
– Nhóm phương thức cung cấp thông tin

+ ...

+ Phương thức cung cấp hoành độ.

+ Phương thức cung cấp tung độ.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Thiết kế lớp CDiem

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



Thiết kế lớp CDim

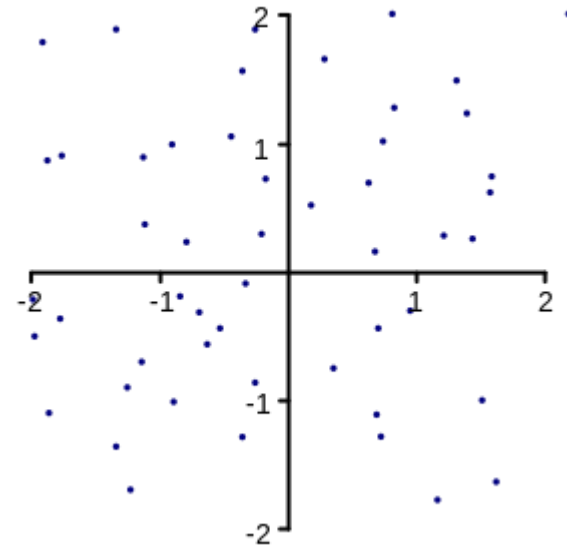
– Nhóm phương thức cập nhật thông tin

+ ...

+ Phương thức cập nhật hoành độ.

+ Phương thức cập nhật tung độ.

+ Kết thúc nhóm phương thức cập nhật thông tin.



Thiết kế lớp CDiem

- 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.
 - + ...
- Tiêu chuẩn so sánh: khoảng cách đến gốc tọa độ (gần gốc tọa độ hơn thì nhỏ hơn).

Thiết kế lớp CDiem

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra điểm trùng gốc không?
 - + Kiểm tra điểm thuộc trục hoành?
 - + Kiểm tra điểm thuộc trục tung?
 - + ...

Thiết kế lớp CDiem

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra điểm có thuộc phần tư thứ I không?

+ Kiểm tra điểm có thuộc phần tư thứ II không?

+ Kiểm tra điểm có thuộc phần tư thứ III không?

+ Kiểm tra điểm có thuộc phần tư thứ IV không?

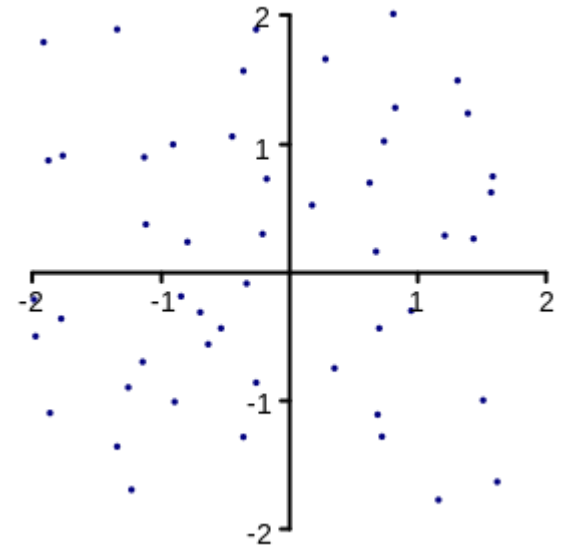
+ ...

Thiết kế lớp CDiem

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai điểm có trùng nhau không?
 - + Kiểm tra hai điểm có ko trùng nhau hay không?
- + Kết thúc nhóm phương thức kiểm tra.

Thiết kế lớp CDim

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



Thiết kế lớp CDiem

— Nhóm phương thức xử lý (tiếp tục)

+ ...

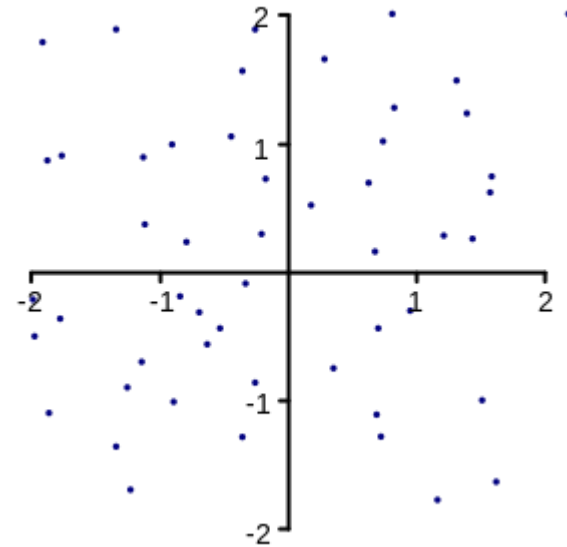
+ Toán tử cộng hai điểm.

+ Toán tử cộng bằng hai điểm.

+ Toán tử trừ hai điểm.

+ Toán tử trừ bằng hai điểm.

+ ...



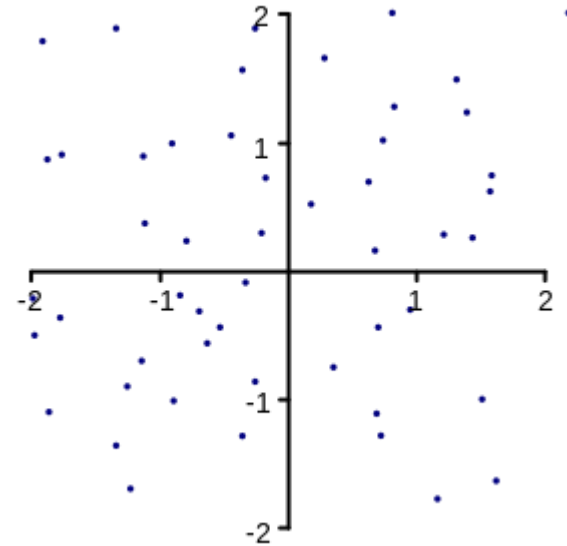
Thiết kế lớp CDiem

— Nhóm phương thức xử lý (tiếp tục)

+ ...

+ Toán tử nhân hai điểm.

+ ...



Thiết kế lớp CDiem

– Nhóm phương thức xử lý (tiếp tục)

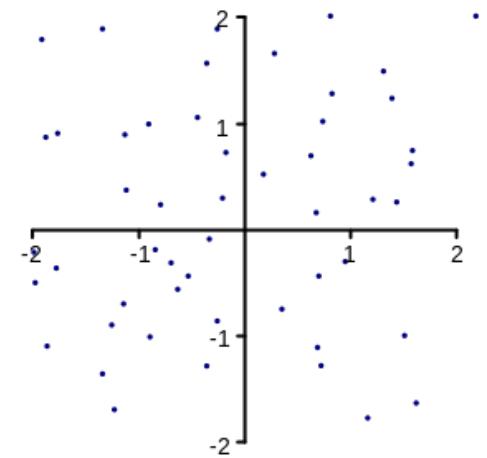
+ ...

+ Phương thức tính khoảng cách đến gốc tọa độ.

+ Phương thức tính khoảng cách đến trục hoành.

+ Phương thức tính khoảng cách đến trục tung.

+ ...



Thiết kế lớp CDiem

– Nhóm phương thức xử lý (tiếp tục)

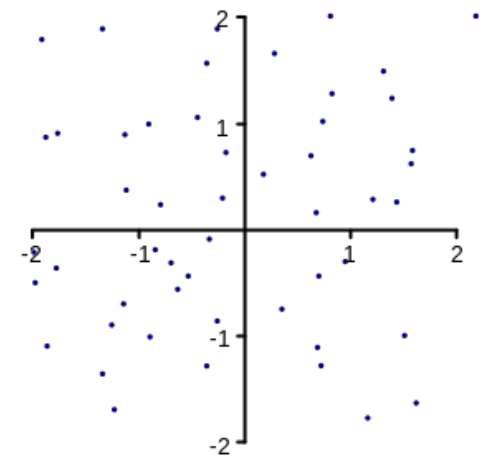
+ ...

+ Tính khoảng cách giữa hai điểm.

+ Tính khoảng cách giữa hai điểm theo phương Ox .

+ Tính khoảng cách giữa hai điểm theo phương Oy .

+ ...



Thiết kế lớp CDiem

— Nhóm phương thức xử lý (tiếp tục)

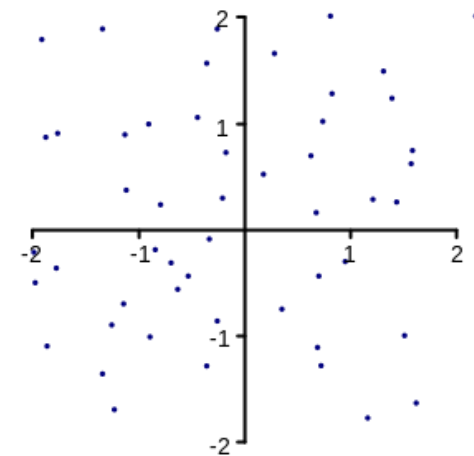
+ ...

+ Tìm tọa độ điểm đối xứng qua gốc tọa độ.

+ Tìm tọa độ điểm đối xứng qua trục hoành.

+ Tìm tọa độ điểm đối xứng qua trục tung.

+ ...



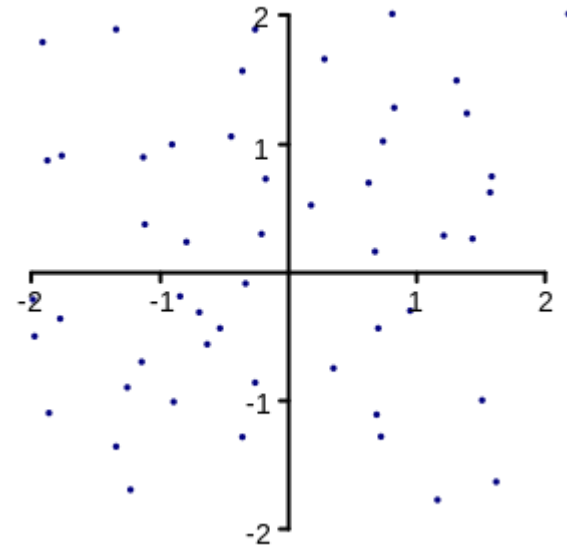
Thiết kế lớp CDiem

– Nhóm phương thức xử lý (tiếp tục)

+ ...

+ Tìm tọa độ điểm đối xứng qua đường phân giác thứ nhất $y = x$.

+ Tìm tọa độ điểm đối xứng qua đường phân giác thứ hai $y = -x$.

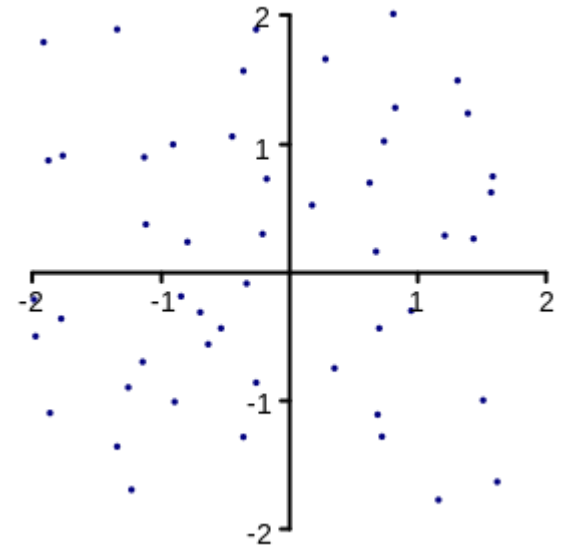


+ Kết thúc nhóm phương thức xử lý.

KHAI BÁO LỚP CDIEM

Thiết kế lớp CDiem

- Thuộc tính
 - + Hoành độ.
 - + Tung độ.

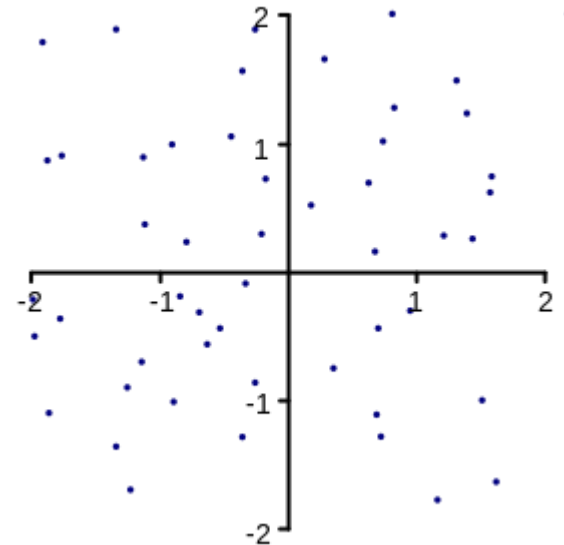


Khai báo lớp CDiem

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

Thiết kế lớp CDiem

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



Khai báo lớp CDiem

```
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&, CDiem&);  
18.     ...
```

Thiết kế lớp CDiem

– Nhóm phương thức khởi tạo (tiếp tục)

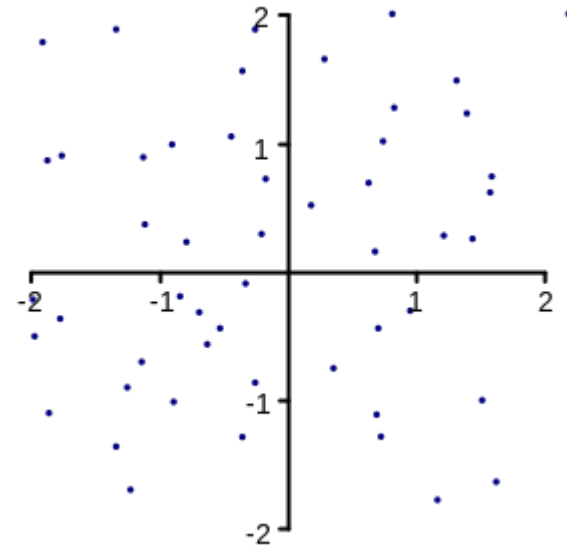
+ ...

+ Phương thức thiết lập mặc định.

+ Phương thức thiết lập sao chép.

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

+ ...



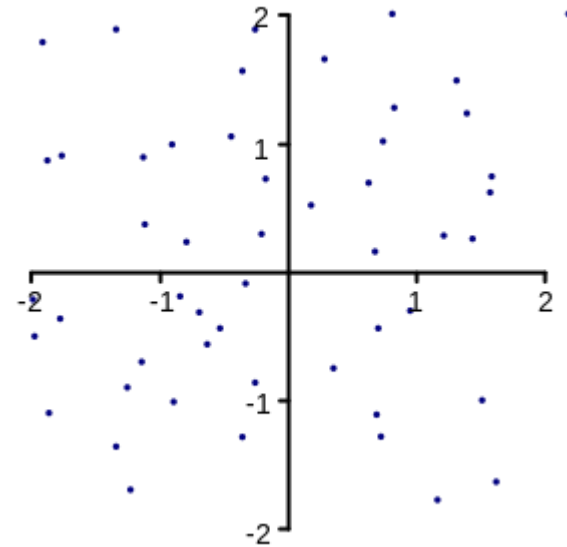
Thiết kế lớp CDiem

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập khi biết hoành độ, tung độ lấy mặc định.

+ ...

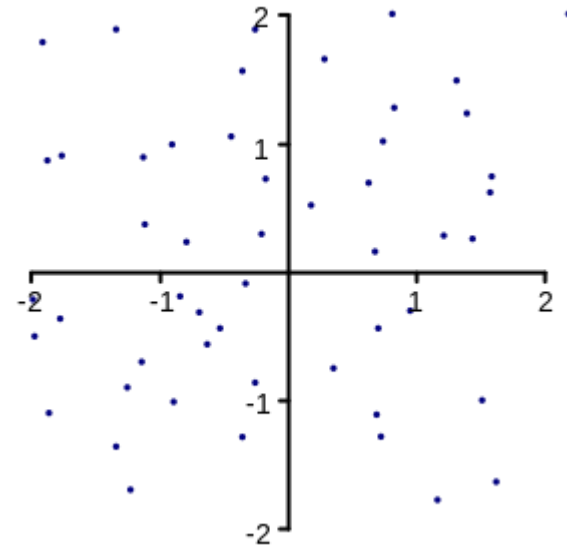


Khai báo lớp CDiem

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


Thiết kế lớp CDiem

- 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.
 - + ...

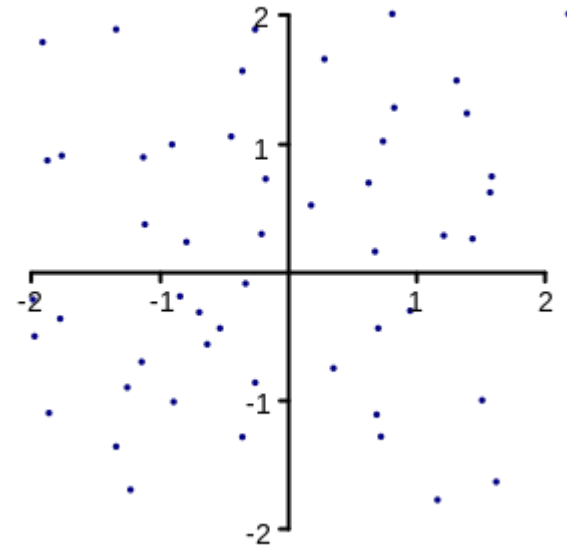


Thiết kế lớp CDiem

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức khởi tạo khi biết hoành độ, tung độ lấy mặc định.



+ Kết thúc nhóm phương thức khởi tạo

Khai báo lớp CDiem

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

Thiết kế lớp CDiem

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



Thiết kế lớp CDiem

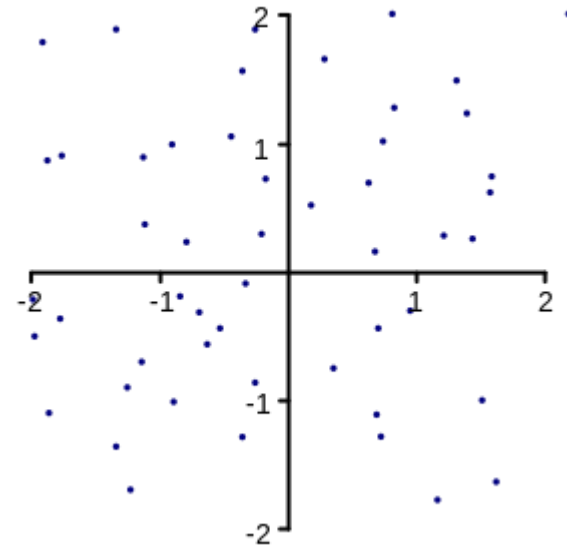
– Nhóm phương thức cung cấp thông tin

+ ...

+ Phương thức cung cấp hoành độ.

+ Phương thức cung cấp tung độ.

+ Kết thúc nhóm phương thức cung cấp thông tin.

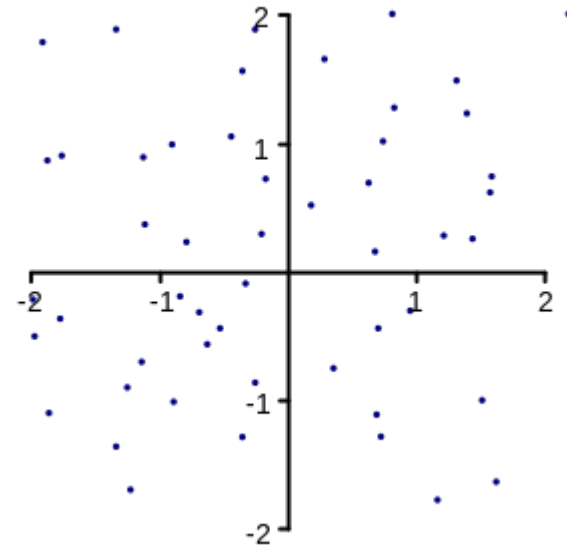


Khai báo lớp CDiem

```
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&, CDiem&);
17. ///// Chương 09 - Thiết kế lớp
18. float getX();
19. float getY();
```

Thiết kế lớp CDiem

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



Thiết kế lớp CDiem

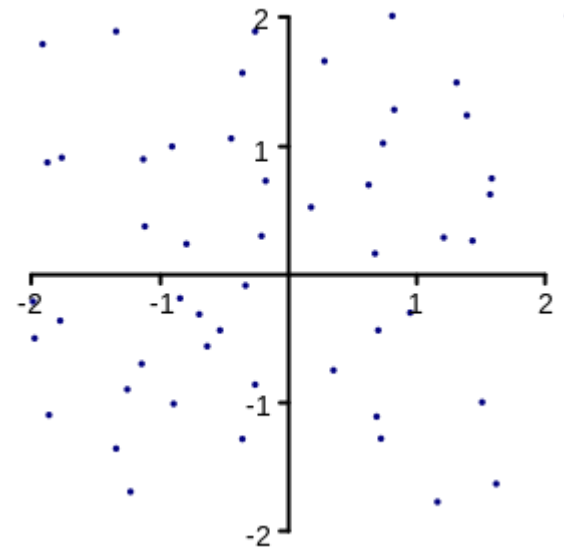
– Nhóm phương thức cập nhật thông tin

+ ...

+ Phương thức cập nhật hoành độ.

+ Phương thức cập nhật tung độ.

+ Kết thúc nhóm phương thức cập nhật thông tin.



Khai báo lớp CDiem

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

Thiết kế lớp CDiem

- 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.
 - + ...
- Tiêu chuẩn so sánh:
khoảng cách đến gốc
tọa độ (gần gốc tọa
độ hơn thì nhỏ hơn).

Khai báo lớp CDiem

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

Thiết kế lớp CDiem

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra điểm trùng gốc không?
 - + Kiểm tra điểm thuộc trục hoành?
 - + Kiểm tra điểm thuộc trục tung?
 - + ...

Khai báo lớp CDiem

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      bool ktTrungGoc();
15.      bool ktThuocHoanh();
16.      bool ktThuocTung();
17.      ...
```

Thiết kế lớp CDiem

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra điểm có thuộc phần tư thứ I không?
 - + Kiểm tra điểm có thuộc phần tư thứ II không?
 - + Kiểm tra điểm có thuộc phần tư thứ III không?
 - + Kiểm tra điểm có thuộc phần tư thứ IV không?
 - + ...

Khai báo lớp CDiem

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      bool ktThuoc1();
15.      bool ktThuoc2();
16.      bool ktThuoc3();
17.      bool ktThuoc4();
18.      ...
```


Thiết kế lớp CDiem

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai điểm có trùng nhau không?
 - + Kiểm tra hai điểm có ko trùng nhau hay không?
- + Kết thúc nhóm phương thức kiểm tra.

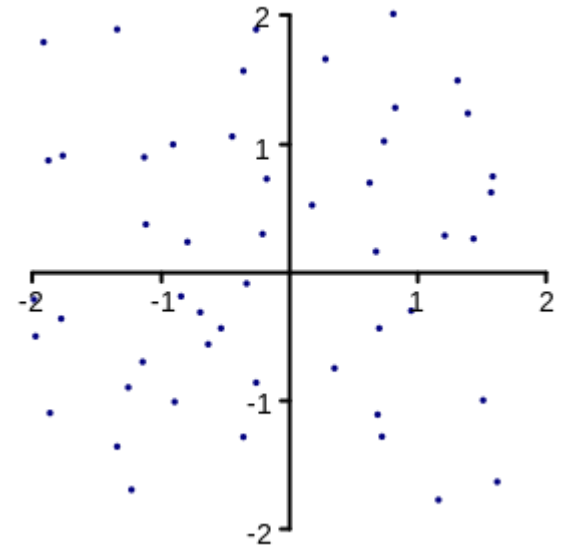
Khai báo lớp CDiem

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      bool ktTrung(const CDiem&);
15.      bool ktKoTrung(const CDiem&);

16.      ///// Nhóm phương thức xử lý
```

Thiết kế lớp CDim

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



Khai báo lớp CDiem

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

Thiết kế lớp CDiem

— Nhóm phương thức xử lý (tiếp tục)

+ ...

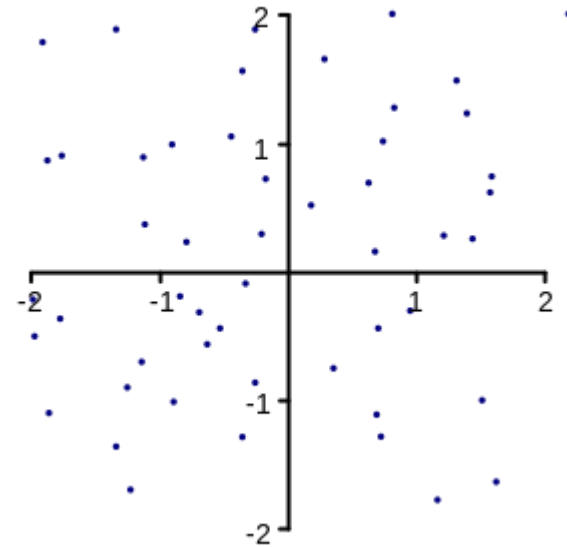
+ Toán tử cộng hai điểm.

+ Toán tử trừ hai điểm.

+ Toán tử cộng bằng hai điểm.

+ Toán tử trừ bằng hai điểm.

+ ...



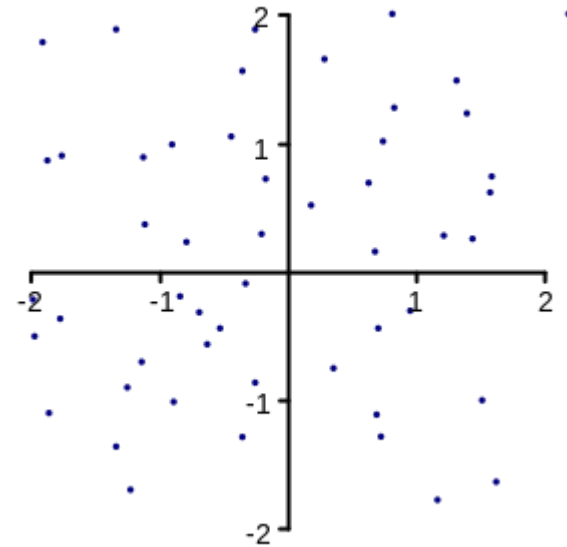
Thiết kế lớp CDiem

— Nhóm phương thức xử lý (tiếp tục)

+ ...

+ Toán tử nhân hai điểm.

+ ...



Khai báo lớp CDiem

```
11.  //... Nhóm phương thức xử lý
12.  //... Chương 07 - Toán tử số học
13.  ...
14.  CDiem operator+(const CDiem&);
15.  CDiem operator-(const CDiem&);
16.  CDiem& operator+=(const CDiem&);
17.  CDiem& operator-=(const CDiem&);
18.  float operator*(const CDiem&);
19.  ...
```

Thiết kế lớp CDiem

– Nhóm phương thức xử lý (tiếp tục)

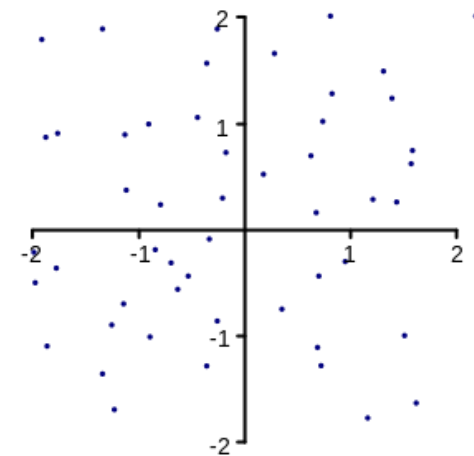
+ ...

+ Phương thức tính khoảng cách đến gốc tọa độ.

+ Phương thức tính khoảng cách đến trục hoành.

+ Phương thức tính khoảng cách đến trục tung.

+ ...



Khai báo lớp CDiem

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


Thiết kế lớp CDiem

– Nhóm phương thức xử lý (tiếp tục)

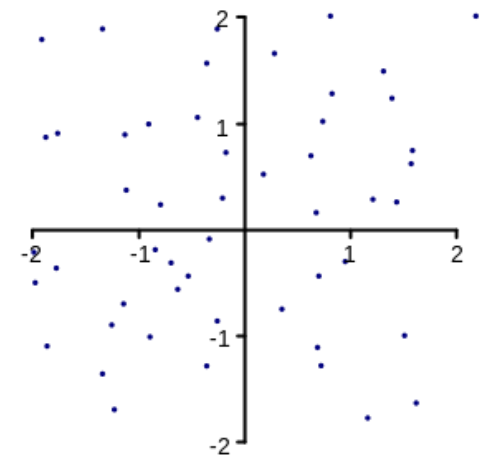
+ ...

+ Tính khoảng cách giữa hai điểm.

+ Tính khoảng cách giữa hai điểm theo phương Ox .

+ Tính khoảng cách giữa hai điểm theo phương Oy .

+ ...



Khai báo lớp CDiem

```
11.  //... Nhóm phương thức xử lý
12.  //... Chương 09 - Thiết kế lớp
13.  ...
14.  float KhoangCach(const CDiem&);
15.  float KhoangCachX(const CDiem&);
16.  float KhoangCachY(const CDiem&);
17.  ...
```

Thiết kế lớp CDiem

— Nhóm phương thức xử lý (tiếp tục)

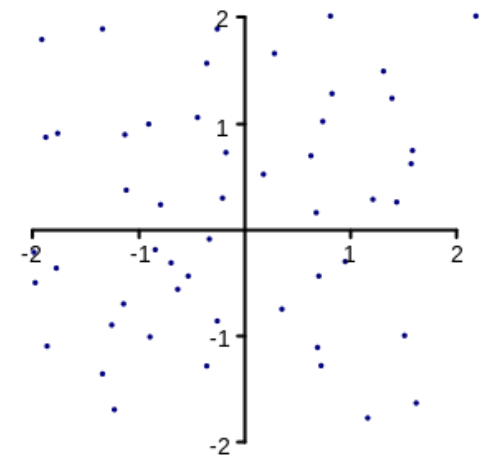
+ ...

+ Tìm tọa độ điểm đối xứng qua gốc tọa độ.

+ Tìm tọa độ điểm đối xứng qua trục hoành.

+ Tìm tọa độ điểm đối xứng qua trục tung.

+ ...



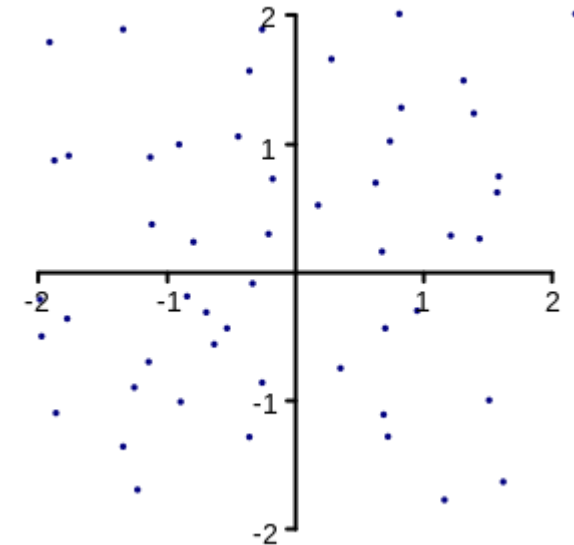
Thiết kế lớp CDiem

– Nhóm phương thức xử lý (tiếp tục)

+ ...

+ Tìm tọa độ điểm đối xứng qua đường phân giác thứ nhất $y = x$.

+ Tìm tọa độ điểm đối xứng qua đường phân giác thứ hai $y = -x$.



+ Kết thúc nhóm phương thức xử lý.

Khai báo lớp CDiem

```
11.      ///// Nhóm phương thức xử lý
12.      ///// Chương 09 - Thiết kế lớp
13.      ...
14.      CDiem DoiXungGoc();
15.      CDiem DoiXungHoanh();
16.      CDiem DoiXungTung();
17.      CDiem DoiXungPhanGiac1();
18.      CDiem DoiXungPhanGiac2();
19. };
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC KHỞI TẠO

Định nghĩa phương thức

```
11. void CDiem::Nhap()  
12. {  
13.     cout << "Nhap x: ";  
14.     cin >> x;  
15.     cout << "Nhap y: ";  
16.     cin >> y;  
17. }
```

Định nghĩa phương thức

```
11.istream& operator>>(istream& is, CDiem& P)
12.{
13.    cout << "Nhap x : ";
14.    is >> P.x;
15.    cout << "Nhap y : ";
16.    is >> P.y;
17.    return is;
18.}
```


Định nghĩa phương thức

```
11. CDiem::CDiem()
```

```
12. {
```

```
13. |   x = 0;
```

```
14. |   y = 0;
```

```
15. }
```

Định nghĩa phương thức

```
11.CDiem::CDiem(const CDiem& P)
```

```
12.{
```

```
13. |   x = P.x;
```

```
14. |   y = P.y;
```

```
15.}
```

Định nghĩa phương thức

```
11.CDiem::CDiem(float xx, float yy)
```

```
12.{
```

```
13. |   x = xx;
```

```
14. |   y = yy;
```

```
15. }
```

Định nghĩa phương thức

```
11. CDiem::CDiem(float xx)
```

```
12. {
```

```
13. |   x = xx;
```

```
14. |   y = 0;
```

```
15. }
```

Định nghĩa phương thức

```
11. void CDiem::KhoiTao()  
12. {  
13. |   x = 0;  
14. |   y = 0;  
15. }
```

Định nghĩa phương thức

```
11. void CDiem::KhoiTao(const CDiem& P)
12. {
13. |   x = P.x;
14. |   y = P.y;
15. }
```

Định nghĩa phương thức

```
11. void CDiem::KhoiTao(float xx, float yy)
12. {
13. |   x = xx;
14. |   y = yy;
15. }
```

Định nghĩa phương thức

```
11. void CDiem::KhoiTao(float xx)
12. {
13.     x = xx;
14.     y = 0;
15. }
```


Định nghĩa các phương thức
**NHÓM PHƯƠNG THỨC
CUNG CẤP THÔNG TIN**

Định nghĩa phương thức

```
11. void CDiem::Xuat()  
12. {  
13. |   cout << "(" << x << ", " << y << ")";  
14. }
```

Định nghĩa phương thức

```
11. ostream& operator<<(ostream& os, CDiem& P)
12. {
13.     os << "(" << P.x << "," << P.y << ")";
14.     return os;
15. }
```

Định nghĩa phương thức

```
11.float CDiem::getX()  
12.{  
13.|    return x;  
14.}
```

Định nghĩa phương thức

```
11.float CDiem::getY()  
12.{  
13.|    return y;  
14.}
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC CẬP NHẬT THÔNG TIN

Định nghĩa phương thức

```
11. CDiem& CDiem::operator=(const CDiem& P)
12. {
13.     x = P.x;
14.     y = P.y;
15.     return *this;
16. }
```

Định nghĩa phương thức

```
11. void CDiem:: setX(float xx)
12. {
13. |     x = xx;
14. }
```


Định nghĩa phương thức

```
11. void CDiem::setY(float yy)
12. {
13. |   y = yy;
14. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC KIỂM TRA

Định nghĩa phương thức

```
11. bool CDiem::operator == (const CDiem& P)
12. {
13.     CDiem temp(P);
14.     if (KhoangCachGoc() == temp.KhoangCachGoc())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDiem::operator != (const CDiem& P)
12. {
13.     CDiem temp(P);
14.     if (KhoangCachGoc() != temp.KhoangCachGoc())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDiem::operator > (const CDiem& P)
12. {
13.     CDiem temp(P);
14.     if (KhoangCachGoc() > temp.KhoangCachGoc())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDiem::operator < (const CDiem& P)
12. {
13.     CDiem temp(P);
14.     if (KhoangCachGoc() < temp.KhoangCachGoc())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDiem::operator >= (const CDiem& P)
12. {
13.     CDiem temp(P);
14.     if (KhoangCachGoc() >= temp.KhoangCachGoc())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CDiem::operator <= (const CDiem& P)
12. {
13.     CDiem temp(P);
14.     if (KhoangCachGoc() <= temp.KhoangCachGoc())
15.         return true;
16.     return false;
17. }
```


Định nghĩa phương thức

```
11. bool CDiem::ktTrungGoc()  
12. {  
13.     if (x == 0 && y == 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktThuocHoanh()  
12. {  
13.     if (y == 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktThuocTung()  
12. {  
13.     if (x == 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktThuoc1()  
12. {  
13.     if (x > 0 && y > 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktThuoc2()  
12. {  
13.     if (x < 0 && y > 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktThuoc3()  
12. {  
13.     if (x < 0 && y < 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktThuoc4()  
12. {  
13.     if (x > 0 && y < 0)  
14.         return true;  
15.     return false;  
16. }
```

Định nghĩa phương thức

```
11. bool CDiem::ktTrung(const CDiem& P)
12. {
13.     if (x == P.x && y == P.y)
14.         return true;
15.     return false;
16. }
```


Định nghĩa phương thức

```
11. bool CDiem::ktKoTrung(const CDiem& P)
12. {
13.     if (!(x == P.x && y == P.y))
14.         return true;
15.     return false;
16. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC XỬ LÝ

Định nghĩa phương thức

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

Định nghĩa phương thức

```
11. CDiem CDiem::operator+(const CDiem& P)
12. {
13.     CDiem temp;
14.     temp.x = x + P.x;
15.     temp.y = y + P.y;
16.     return temp;
17. }
```

Định nghĩa phương thức

```
11.CDiem CDiem::operator-(const CDiem& P)
12.{
13.    CDiem temp;
14.    temp.x = x - P.x;
15.    temp.y = y - P.y;
16.    return temp;
17.}
```

Định nghĩa phương thức

```
11.float CDiem::operator*(const CDiem& P)
12.{
13.|    return x * P.x + y * P.y;
14.}
```

Định nghĩa phương thức

```
11. CDiem& CDiem::operator+=(const CDiem& P)
12. {
13.     x += P.x;
14.     y += P.y;
15.     return *this;
16. }
```

Định nghĩa phương thức

```
11. CDiem& CDiem::operator-=(const CDiem& P)
12. {
13.     x -= P.x;
14.     y -= P.y;
15.     return *this;
16. }
```


Định nghĩa phương thức

```
11.float CDiem::KhoangCachGoc()  
12.{  
13.|    return sqrt(x * x + y * y);  
14.}
```

Định nghĩa phương thức

```
11.float CDiem::KhoangCachHoanh()  
12.{  
13.|    return abs(y);  
14.}
```

Định nghĩa phương thức

```
11.float CDiem::KhoangCachTung()  
12.{  
13.|    return abs(x);  
14.}
```

Định nghĩa phương thức

```
11.float CDiem::KhoangCach(const CDiem& P)
12.{
13.    return sqrt((x - P.x) * (x - P.x) +
14.                (y - P.y) * (y - P.y));
15.}
```

Định nghĩa phương thức

```
11.float CDiem::KhoangCachX(const CDiem& P)
12.{
13.|   return abs(x - P.x);
14.}
```

Định nghĩa phương thức

```
11.float CDiem::KhoangCachY(const CDiem& P)
12.{
13.|   return abs(y - P.y);
14.}
```

Định nghĩa phương thức

```
11. CDiem CDiem::DoiXungGoc()  
12. {  
13.     CDiem temp;  
14.     temp.x = -x;  
15.     temp.y = -y;  
16.     return temp;  
17. }
```

Định nghĩa phương thức

```
11. CDiem CDiem::DoiXungHoanh()  
12. {  
13.     CDiem temp;  
14.     temp.x = x;  
15.     temp.y = -y;  
16.     return temp;  
17. }
```


Định nghĩa phương thức

```
11. CDiem CDiem::DoiXungTung()  
12. {  
13.     CDiem temp;  
14.     temp.x = -x;  
15.     temp.y = y;  
16.     return temp;  
17. }
```

Định nghĩa phương thức

```
11. CDiem CDiem::DoiXungPhanGiac1()  
12. {  
13. |   CDiem temp;  
14. |   temp.x = y;  
15. |   temp.y = x;  
16. |   return temp;  
17. }
```

Định nghĩa phương thức

```
11. CDiem CDiem::DoiXungPhanGiac2()  
12. {  
13.     CDiem temp;  
14.     temp.x = -y;  
15.     temp.y = -x;  
16.     return temp;  
17. }
```

Cảm ơn quý vị đã lắng nghe

Nhóm tác giả

TS. Nguyễn Tấn Trần Minh Khang