



Thiết kế lớp đường thẳng trong mặt phẳng Oxy

1. ThS. Nguyễn Hữu Lợi
2. ThS. Nguyễn Văn Toàn
3. TS. Nguyễn Duy Khánh
4. TS. Nguyễn Tấn Trần Minh Khang

Thiết kế lớp đối tượng đường thẳng CDuongThang



- Thuộc tính
 - + Hệ số a.
 - + Hệ số b.
 - + Hệ số c.



Thiết kế lớp đối tượng đường thẳng CDuongThang



```
11.class CDuongThang
12.{
13.    private:
14.        float a;
15.        float b;
16.        float c;
17.    public:
```



Thiết kế lớp đối tượng đường thẳng

CDuongThang



- Thuộc tính
 - + Hệ số a.
 - + Hệ số b.
 - + Hệ số c.
- Phương thức
 - + Nhóm phương thức khởi tạo.
 - + Nhóm phương thức cung cấp thông tin.
 - + Nhóm phương thức cập nhật thông tin.
 - + Nhóm phương thức xử lý.
 - + Nhóm phương thức kiểm tra.



Thiết kế lớp đối tượng đường thẳng



```
11.class CDuongThang
12.{
13.    private:
14.        float a;
15.        float b;
16.        float c;
17.    public:
18.        // Nhóm phương thức khởi tạo
19.        // Nhóm phương thức cung cấp thông tin
20.        // Nhóm phương thức cập nhật thông tin
21.        // Nhóm phương thức kiểm tra
22.        // Nhóm phương thức xử lý
```

Lớp đối tượng đường thẳng

CDuongThang



— Nhóm phương thức khởi tạo

- + Phương thức khởi tạo mặc định.
- + Phương thức khởi tạo sao chép.
- + Phương thức khởi tạo khi biết đầy đủ thông tin.
- + Phương thức thiết lập mặc định.
- + Phương thức thiết lập sao chép.
- + Phương thức thiết lập khi biết đầy đủ thông tin.
- + Phương thức Nhập.
- + Toán tử vào.

Thiết kế lớp đối tượng đường thẳng

CDuongThang



```
17. // Nhóm phương thức khởi tạo
18. void KhoiTao();
19. void KhoiTao(float, float, float);
20. void KhoiTao(const CDuongThang&);
21. CDuongThang();
22. CDuongThang(float, float, float);
23. CDuongThang(const CDuongThang&);
24. void Nhap();
25. friend istream&operator>>(istream&, CDuongThang&);
```

Lớp đối tượng đường thẳng

CDuongThang



– Nhóm phương thức cung cấp thông tin

+ Phương thức Xuất.

+ Toán tử ra.

+ Phương thức cung cấp hệ số a.

+ Phương thức cung cấp hệ số b.

+ Phương thức cung cấp hệ số c.



Thiết kế lớp đối tượng đường thẳng

CDuongThang



```
26. // Nhóm phương thức cung cấp thông tin
27. void Xuat();
28. friend ostream&operator<<(ostream&, CDuongThang&);
29. float getA();
30. float getB();
31. float getC();
```



Lớp đối tượng đường thẳng

CDuongThang



- Nhóm phương thức cập nhật thông tin
 - + Toán tử gán.
 - + Phương thức cập nhật Hệ số a.
 - + Phương thức cập nhật Hệ số b.
 - + Phương thức cập nhật Hệ số c.



Thiết kế lớp đối tượng đường thẳng CDuongThang



```
31. // Nhóm phương thức cập nhật thông tin
32. CDuongThang& operator = (const CDuongThang&);
33. void setA(float);
34. void setB(float);
35. void setC(float);
```



Lớp đối tượng đường thẳng

CDuongThang



— Nhóm phương thức kiểm tra

- + Kiểm tra điểm có thuộc đường thẳng không?
- + Kiểm tra hai đường thẳng có trùng nhau hay không?
- + Kiểm tra hai đường thẳng có song song với nhau không?
- + Kiểm tra hai đường thẳng có cắt nhau không?
- + Kiểm tra hai đường thẳng có vuông góc không?

Thiết kế lớp đối tượng đường thẳng

CDuongThang



```
35. // Nhóm phương thức kiểm tra
36. int ktThuoc(CDiem);
37. int ktTrung(CDuongThang);
38. int ktSongSong(CDuongThang);
39. int ktCat(CDuongThang);
40. int ktVuongGoc(CDuongThang);
```


Lớp đối tượng đường thẳng

CDuongThang



- Nhóm phương thức xử lý
 - + Toán tử so sánh bằng
 - + Toán tử so sánh khác

Thiết kế lớp đối tượng đường thẳng

CDuongThang



```
42. | // Nhóm phương thức xử lý
43. | int operator == (const CDuongThang&);
44. | int operator != (const CDuongThang&);
```

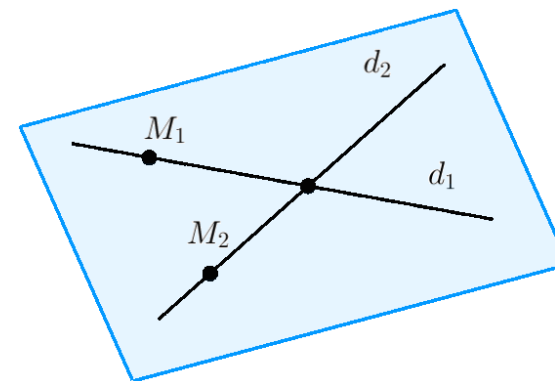
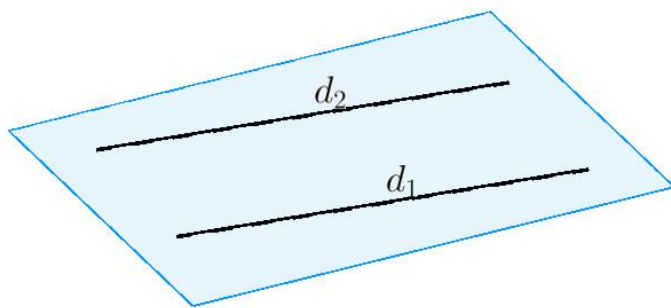
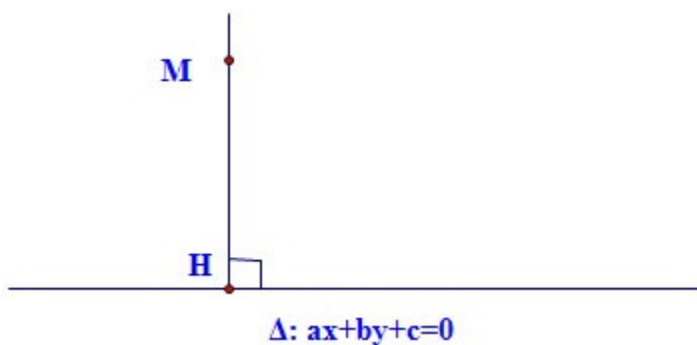
Lớp đối tượng đường thẳng

CDuongThang



– Nhóm phương thức xử lý

- + Tính khoảng cách giữa điểm và đường thẳng
- + Tính khoảng cách giữa hai đường thẳng song song
- + Tìm tọa độ điểm giao nhau giữa hai đường thẳng (nếu có)
- + Phương thức phá hủy

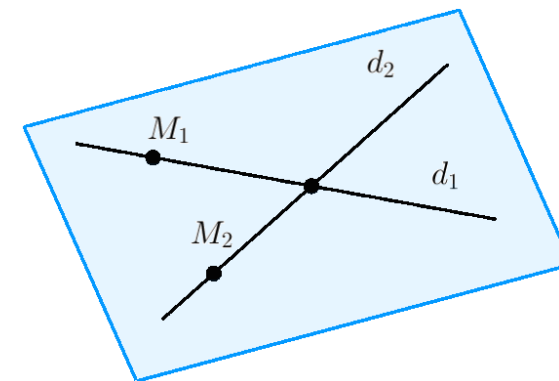
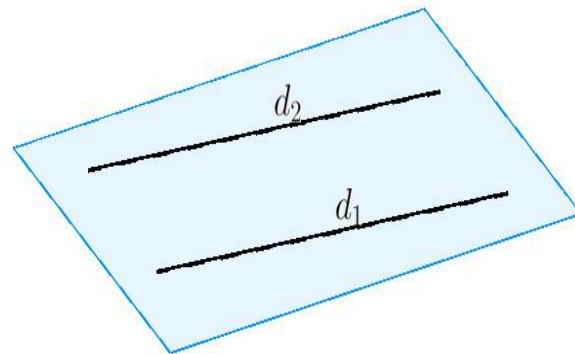
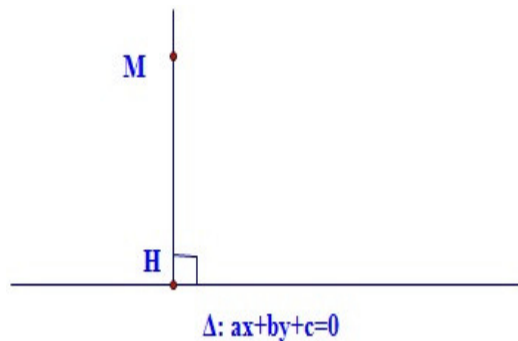


Thiết kế lớp đối tượng đường thẳng

CDuongThang



```
49. // Nhóm phương thức xử lý
50. float KhoangCach(CDiem);
51. float KhoangCach(CDuongThang);
52. CDiem GiaoDiem(CDuongThang);
53. ~DuongThang();
```



9.. Thiết kế lớp đường thẳng



- Định nghĩa các phương thức cung cấp thông tin.

- Cách 01.

```
11.float CDuongThang::getA()
```

```
12.{
```

```
13.|    return a;
```

```
14.}
```

- Cách 02.

```
11.float CDuongThang::getA()
```

```
12.{
```

```
13.|    return this->a;
```

```
14.}
```

Bên trong thân phương thức của một lớp đối tượng, **this** là một con trỏ đối tượng thuộc về lớp mà phương thức đó thuộc về, con trỏ đối tượng **this** giữ địa chỉ của đối tượng đang gọi thực hiện phương thức. Hơn nữa, ***this** chính là đối tượng đang gọi thực hiện phương thức.

9.. Thiết kế lớp đường thẳng



- Định nghĩa các phương thức cung cấp thông tin.

- Cách 01.

```
11.float CDuongThang::getB()
```

```
12.{
```

```
13.|    return b;
```

```
14.}
```

- Cách 02.

```
11.float CDuongThang::getB()
```

```
12.{
```

```
13.|    return this->b;
```

```
14.}
```

Bên trong thân phương thức của một lớp đối tượng, **this là một con trỏ đối tượng thuộc về lớp mà phương thức đó thuộc về**, con trỏ đối tượng this giữ địa chỉ của đối tượng đang gọi thực hiện phương thức. Hơn nữa, ***this chính là đối tượng đang gọi thực hiện phương thức.**

9.. Thiết kế lớp đường thẳng



- Định nghĩa các phương thức cung cấp thông tin.

- Cách 01.

```
11.float CDuongThang::getC()
```

```
12.{
```

```
13. |    return c;
```

```
14.}
```

- Cách 02.

```
11.float CDuongThang::getC()
```

```
12.{
```

```
13. |    return this->c;
```

```
14.}
```

Bên trong thân phương thức của một lớp đối tượng, **this** là một con trỏ đối tượng thuộc về lớp mà phương thức đó thuộc về, con trỏ đối tượng **this** giữ địa chỉ của đối tượng đang gọi thực hiện phương thức. Hơn nữa, ***this** chính là đối tượng đang gọi thực hiện phương thức.

9.. Thiết kế lớp đường thẳng³



```
11. void CDuongThang::setA(float aa)
12. {
13.     a = aa;
14. }
15. void CDuongThang::setB(float bb)
16. {
17.     b = bb;
18. }
19. void CDuongThang::setC(float cc)
20. {
21.     c = cc;
22. }
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức kiểm tra.

```
11.int CDuongThang::ktThuoc(CDiem P)
12.{
13.    if ((a * P.getX() + b * P.getY() + c) == 0)
14.        return 1;
15.    return 0;
16.}
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức kiểm tra.

```
11.int CDuongThang::ktTrung(CDuongThang d)
12.{
13.    if (a == d.a && b == d.b && c == d.c)
14.        return 1;
15.    return 0;
16.}
```


9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức kiểm tra.

```
11.int CDuongThang::ktSongSong(CDuongThang d)
12.{
13.    if (a == d.a && b == d.b && c != d.c)
14.        return 1;
15.    return 0;
16.}
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức kiểm tra.

```
11.int CDuongThang::ktCat(CDuongThang d)
12.{
13.    float D = a * d.b - d.a * b;
14.    if (D != 0)
15.        return 1;
16.    return 0;
17.}
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức kiểm tra.

```
11.int CDuongThang::ktVuongGoc(CDuongThang d)
12.{
13.    if ((a * d.a + b * d.b) == 0)
14.        return 1;
15.    return 0;
16.}
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
11. void CDuongThang::KhoiTao()  
12. {  
13.     a = 0;  
14.     b = 0;  
15.     c = 0;  
16. }
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
11. void CDuongThang::KhoiTao(float aa)
12. {
13.     a = aa;
14.     b = 0;
15.     c = 0;
16. }
```


9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
1. CDuongThang::CDuongThang(float aa, float bb, float cc)
2. {
3.     a = aa;
4.     b = bb;
5.     c = cc;
6. }
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
11. void CDuongThang::KhoiTao(float aa, float bb, float cc)
12. {
13.     a = aa;
14.     b = bb;
15.     c = cc;
16. }
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
1. void CDuongThang::KhoiTao(const CDuongThang& d)
2. {
3.     a = d.a;
4.     b = d.b;
5.     c = d.b;
6. }
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

1. CDuongThang::CDuongThang()

2. {

3. | a = 0;

4. | b = 0;

5. | c = 0;

6. }

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

1. `CDuongThang::CDuongThang(float aa)`

2. `{`

3. `a = aa;`

4. `b = 0;`

5. `c = 0;`

6. `}`

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
1. CDuongThang::CDuongThang(float aa, float bb, float cc)
2. {
3.     a = aa;
4.     b = bb;
5.     c = cc;
6. }
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức khởi tạo.

```
1. CDuongThang::CDuongThang(const CDuongThang& d)
2. {
3.     a = d.a;
4.     b = d.b;
5.     c = d.b;
6. }
```


9.. Thiết kế lớp đường thẳng



```
11. void CDuongThang::Nhap()  
12. {  
13.     cout << "Nhap he so a: ";  
14.     cin >> a;  
15.     cout << "\nNhap he so b: ";  
16.     cin >> b;  
17.     cout << "\nNhap he so c: ";  
18.     cin >> c;  
19. }
```

9.. Thiết kế lớp đường thẳng



```
11. void CDuongThang::Xuat()  
12. {  
13. |   cout << a << "x + " << b << "y + " << c << " = 0" << endl;  
14. }
```

9.. Thiết kế lớp đường thẳng



```
11.istream& operator>>(istream& is, CDuongThang& d)
12.{
13.    cout << "    + Nhap he so a: ";
14.    is >> d.a;
15.    cout << "    + Nhap he so b: ";
16.    is >> d.b;
17.    cout << "    + Nhap he so c: ";
18.    is >> d.c;
19.    return is;
20.}
```

9.. Thiết kế lớp đường thẳng



```
11 ostream& operator<<(ostream& os, CDuongThang& d)
12 {
13     |   os << d.a << "x + " << d.b << "y + " << d.c << " = 0";
14     |   return os;
15 }
```

9.. Thiết kế lớp đường thẳng



```
11.int CDuongThang::operator == (const CDuongThang& x)
12.{
13.    if (a == x.a && b == x.b && c == x.c)
14.        return 1;
15.    return 0;
16.}
```

9.. Thiết kế lớp đường thẳng



```
11.int CDuongThang::operator != (const CDuongThang& x)
12.{
13.    if (a != x.a || b != x.b || c != x.c)
14.        return 1;
15.    return 0;
16.}
```

9.. Thiết kế lớp đường thẳng



```
11.float CDuongThang::KhoangCach(CDiem P)
12.{
13.    float tu = abs(a * P.getX() + b * P.getY() + c);
14.    float mau = sqrt(a * a + b * b);
15.    return tu / mau;
16.}
```


9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức xử lý

```
11.float CDuongThang::KhoangCach(CDuongThang d)
12.{
13.    float a = 0;
14.    if (ktSongSong(d) == 1)
15.    {
16.        //Nhập điểm thuộc đường thẳng
17.        CDiem P;
18.        P.NhapDiem();
```

9.. Thiết kế lớp đường thẳng



— Định nghĩa các phương thức xử lý

```
11. |         if (ktThuoc(P) == 1)
12. |             a = d.KhoangCach(P);
13. |     return a;
14. }
```



Cảm ơn quý vị đã lắng nghe

Nhóm tác giả

TS. Nguyễn Tấn Trần Minh Khang