

Geometria Computacional

Ponto

Todas as notações de double podem ser adaptadas a depender do problema

- Classe Ponto

```
template<typename T>
struct Point{
    T x;
    T y;

    bool operator <(const Point& a){
        return x==a.x ? y<a.y : x<a.x;
    }
    bool operator ==(const Point& a){
        return x==a.x && y==a.y;
    }
    double distance(Point a){
        return hypot(x-a.x, y-a.y);
    }
};
```

Reta

A equação da reta mostrada abaixo se refere a $ax+by+c=0$

- Classe reta

```
template<typename T>
struct Line{
    T a;
    T b;
    T c;

    T fx(T x){
        return -(a*x+c)/b;
    }

    T fy(T y){
        return -(b*y+c)/a;
    }
};
```

Criação de uma reta a partir de dois pontos

```
Line new_line(Point p, Point q){
    Line ans;
    ans.a = p.y-q.y;
    ans.b = q.x-p.x;
    ans.c = p.x*p.y-q.x*p.y;

    return ans;
};
```

Círculo

- Classe círculo

As funções definidas abaixo se referem ao arco, corda, setor e segmento do círculo respectivamente.

```
const double PI = acos(-1);

struct Circle{
    double r;
    double x;
    double y;

    double arc(double ang){
        return ang*r;
    }

    double chord(double ang){
        return 2*PI*sin(ang/2);
    }

    double sector(double ang){
        return ang*r*r/2;
    }

    double segment(double ang){
        double s = sector(ang);
        double c = chord(ang);
        return sqrt((s-r)*(s-r)*(s-c));
    }
};
```