

Examen de Programmation C++

Durée : 1h

Aucun document autorisé

Mobiles interdits

Nom :

Prénom :

On souhaite représenter un point $p = (x, y)$ du plan cartésien.

► 1. Écrivez en C++ la classe `Point` avec :

- deux variables privées `x` et `y` de type `double`;
- un ou plusieurs constructeurs pour initialiser l'objet courant;
- les accesseurs `getX` et `getY`;
- les mutateurs `setX` et `setY`;
- la fonction `distance` qui renvoie la distance entre le point courant et un point passé en paramètre. Rappel : la distance entre 2 points $p_1 = (x_1, y_1)$ et $p_2 = (x_2, y_2)$ est $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$;
- la surcharge de l'opérateur `<<` pour écrire sur un `ostream` un point au format (x, y) .

```
#ifndef POINT_H
#define POINT_H

#include <iostream>
#include <cmath>

class Point {

private:
    double x, y;

public:
    // le constructeur
    Point(double a=0, double b=0) : x(a), y(b) {}
    // accesseurs
    inline double getX() const { return this->x; }
    inline double getY() const { return this->y; }
    // les mutateurs
    inline void setX(double x) { this->x = x; }
    inline void setY(double y) { this->y = y; }
    // Rôle : renvoie la distance entre this et p
    double distance(const Point &p) const {
        double dx = p.getX()-this->getX();
        double dy = p.getY()-this->getY();
        return sqrt(dx*dx+dy*dy);
    }
}
```

```
// Rôle : écrit le Point p sur le flot f
friend std::ostream&operator<<(std::ostream &f, const Point& p) {
    return f << "(" << p.getX() << ", " << p.getY() << ")";
}
};

#endif
```

► 2. Écrivez la fonction `main` qui :

1. déclare et construit les 2 points $p_1 = (3.2, -7.89)$ et $p_2 = (-9.56, 0.0)$. Le premier point p_1 sera alloué dynamiquement, mais pas le second.
2. puis écrit la sortie standard les valeurs des ces 2 points et la distance qui les sépare. Avec les 2 points précédents, on obtiendra quelque chose comme :
la distance entre les points (3.2,-7.89) et (-9.56,0) est égale à 15.0023

```
#include "Point.hpp"
#include <iostream>
#include <cstdlib>

int main() {
    Point *p1 = new Point(3.2, -7.89);
    Point p2(-9.56);

    std::cout << "la distance entre les points " << *p1 << " et " << p2;
    std::cout << " est égale à " << p1->distance(p2) << std::endl;

    return EXIT_SUCCESS;
}
```