

```

1  #include <iostream>
2  #include <string>
3  #include <math.h>
4  #include <cstdio>
5  #include "Point2D.hpp"
6  using namespace std;
7  #define MAX_LENGTH 20
8
9  // Costruttori
10 Point2D::Point2D(float a, float b){
11     x = a;
12     y = b;
13     i = -1;
14 }
15 Point2D::Point2D(float a, float b, int j){
16     x = a;
17     y = b;
18 }
19 Point2D::Point2D(){
20     x = 0;
21     y = 0;
22     i = -1;
23 }
24
25 // Distruttore
26 Point2D::~~Point2D(){ }
27
28 // Costruttore di copia
29 Point2D::Point2D(const Point2D &point){
30     x = point.x;
31     y = point.y;
32     i = point.i;
33 }
34
35 // Ridefinizione degli operatori di somma e differenza
36 Point2D Point2D::operator+ (Point2D &a){
37     return Point2D(x+a.x, y+a.y);
38 }
39 Point2D Point2D::operator- (Point2D &a){
40     return Point2D(x-a.x, y-a.y);
41 }
42
43 // Metodi getter
44 int Point2D::get_i(){
45     return i;
46 }
47 float Point2D::get_x(){
48     return x;
49 }
50 float Point2D::get_y(){
51     return y;
52 }
53
54 // Metodi setter
55 void Point2D::set_x(float a){
56     x = a;
57 }
58 void Point2D::set_y(float b){
59     y = b;
60 }
61 void Point2D::set_i(int j){
62     i = j;
63 }
64
65 // Metodo per il calcolo della norma del punto
66 float Point2D::norm(){

```

```
67     return sqrt(x*x+y*y);
68 }
69
70 // Metodo che restituisce una stringa con le coordinate del punto tra parentesi tonde
71 char* Point2D::toString() {
72     char* s = new char[MAX_LENGTH];
73     sprintf(s, "(%.3f, %.3f) ", x,y);
74     return s;
75 }
76
```