Polymorphism_Lab2

May 12, 2020

In [ ]:
```
#include <assert.h>

// TODO: Define Point class
class Point{
public:
    // TODO: Define public constructor
    Point(int a=0, int b=0):x(a), y(b){}
    // TODO: Define + operator overload
    Point operator+(const Point &p){
        Point sum;
        sum.x = x + p.x;
        sum.y = y + p.y;
        return sum;
    }
    // TODO: Declare attributes x and y
    int x;
    int y;
};

// Test in main()
int main() {
  Point p1(10, 5), p2(2, 4);
  Point p3 = p1 + p2; // An example call to "operator +";
  assert(p3.x == p1.x + p2.x);
  assert(p3.y == p1.y + p2.y);
}
```

Compile & Execute

Hide Solution

Explain

In [ ]:
```
#include <assert.h>

// TODO: Define Point class
class Point {
```

```
public:
  // TODO: Define public constructor
  Point(int x = 0, int y = 0) : x(x), y(y) {}

  // TODO: Define + operator overload
  Point operator+(const Point& addend) {
    Point sum;
    sum.x = x + addend.x;
    sum.y = y + addend.y;
    return sum;
  }

  // TODO: Declare attributes x and y
  int x, y;
};

// Test in main()
int main() {
  Point p1(10, 5), p2(2, 4);
  Point p3 = p1 + p2; // An example call to "operator +";
  assert(p3.x == p1.x + p2.x);
  assert(p3.y == p1.y + p2.y);
}
```

Loading terminal (id_ovjrzbc), please wait...