Đã bắt đầu vào lúc	Thứ hai, 25 Tháng chín 2023, 7:15 AM
Tình trạng	Đã hoàn thành
Hoàn thành vào lúc	Thứ năm, 28 Tháng chín 2023, 2:02 AM
Thời gian thực hiện	2 ngày 18 giờ
Điểm	11,00/11,00
Điểm	**10,00** của 10,00 (**100**%)

Câu hỏi 1

Chính xác

Điểm 1,00 của 1,00

In the coordinate plane, we have class Point to store a point with it's x-y coordinate.

Your task in this exercise is to implement functions marked with /* * STUDENT ANSWER */.

Note: For exercises in Week 1, we have #include <bits/stdc++.h> and using namespace std;

For example:

Test	Result
`Point A(2, 3);` `cout << A.getX() << " " << A.getY();`	`2 3`

`Point A(2, 3);`
`Point B(1, 1);`
`cout << pow(A.distanceToPoint(B), 2);`
	`5`

Answer: (penalty regime: 0 %)

Reset answer

```
        this->x = x;
    }

    void setY(double y)
    {
        /*
         * STUDENT ANSWER
         */
        this->y = y;
    }

    double getX() const
```

```
    {
        /*
         * STUDENT ANSWER
         */
        return this->x;
    }

    double getY() const
    {
        /*
         * STUDENT ANSWER
         */
        return this->y;
    }

    double distanceToPoint(const Point &pointA)
    {
        /*
         * STUDENT ANSWER
         * TODO: calculate the distance from this point to point A in the coordin
         */
        return sqrt(pow(this->x - pointA.x, 2) + pow(this->y - pointA.y, 2));
    }
};
```

	Test	Expected	Got	
✔	`Point A(2, 3);` `cout << A.getX() << " " << A.getY();`	2 3	2 3	✔

✔	`Point A(2, 3);`
	`Point B(1, 1);`
	`cout << pow(A.distanceToPoint(B), 2);`
		5	5	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 2

Chính xác

Điểm 1,00 của 1,00

In the coordinate plane, a circle is defined by center and radius.

Your task in this exercise is to implement functions marked with /* * STUDENT ANSWER */.

Note: you can use implemented class Point in *previous question*

For example:

Test	Result
`Circle A;` `A.printCircle();`	`Center: {0.00, 0.00} and Radius 0.00`

Answer: (penalty regime: 0 %)

Reset answer

```
106     void setCenter(Point point)
107     {
108         /*
109          * STUDENT ANSWER
110          */
111         this->center.setX(point.getX());
112         this->center.setY(point.getY());
113     }
114 
115     void setRadius(double radius)
116     {
117         /*
118          * STUDENT ANSWER
119          */
120         this->radius = radius;
121     }
```

```
122
123     Point getCenter() const
124     {
125         /*
126          * STUDENT ANSWER
127          */
128         return this->center;
129     }
130
131     double getRadius() const
132     {
133         /*
134          * STUDENT ANSWER
135          */
136         return this->radius;
137     }
138
139     void printCircle()
140     {
141         printf("Center: {%.2f, %.2f} and Radius %.2f\n", this->center.getX(), th
142
```

	Test	Expected	Got	
✔	Circle A; A.printCircle();	Center: {0.00, 0.00} and Radius 0.00	Center: {0.00, 0.00} and Radius 0.00	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 3

Chính xác

Điểm 1,00 của 1,00

In this exercise, you can use implemented functions in *previous question* (if needed) and implement these following functions.

```
bool containsPoint(const Point point){}
```

```
bool containsTriangle(const Point pointA, const Point pointB, const Point pointC){}
```

For example:

Test	Result
Point pointO(0, 2); Point point1(1, 2); Circle A = Circle(pointO, 2); cout << A.containsPoint(point1);	1
Point pointO(0, 0); Point point1(1, 0), point2(-1, 0), point3(0, 3); Circle A = Circle(pointO, 3); cout << A.containsTriangle(point1, point2, point3);	0

Answer: (penalty regime: 0 %)

Reset answer

```
51
52      double getY() const
53      {
```

```
         * STUDENT ANSWER
         */
        return this->y;
    }

    double distanceToPoint(const Point &pointA)
    {
        /*
         * STUDENT ANSWER
         * TODO: calculate the distance from this point to point A in the coordin
         */
        return sqrt(pow(this->x - pointA.x, 2) + pow(this->y - pointA.y, 2));
    }
};

class Circle
{
private:
    Point center;
    double radius;

public:
    Circle()
    {
        /*
         * STUDENT ANSWER
         * TODO: set zero center's x-y and radius
         */
        center.setX(0);
        center.setY(0);
        this->radius = 0;
    }
```

	Test	Expected	Got	

✔	`Point pointO(0, 2);`
	`Point point1(1, 2);`
	`Circle A = Circle(pointO, 2);`
	`cout << A.containsPoint(point1);`
		1	1	✔

✔	`Point pointO(0, 0);`
	`Point point1(1, 0), point2(-1, 0), point3(0, 3);`
	`Circle A = Circle(pointO, 3);`
	`cout << A.containsTriangle(point1, point2, point3);`
		0	0	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 4

Chính xác

Điểm 1,00 của 1,00

In this exercise, you can use implemented functions in *previous question* (if needed) and implement these following functions.

1. Overload operator =
2. Overload operator == (The two circles are equal if they have the same center and radius)
3. Overload operator >> (stdin center.x, center.y, radius in order)

For example:

Test	Input	Result

`Point pointO(0, 0);`
`Circle A = Circle(pointO, 3);`
`Circle B;`
`B = A;`
`cout << (B == A);`
		`1`

`Circle A;`
`cin >> A;`
`A.printCircle();`
	`2`
	`3.5`
	`2`
		`Center: {2.00, 3.50} and Radius 2.00`

Answer: (penalty regime: 0 %)

Reset answer

```
26      }
27
28      void setX(double x)
29      {
30          /*
31           * STUDENT ANSWER
```

```
32          */
33          this->x = x;
34      }
35
36      void setY(double y)
37      {
38          /*
39           * STUDENT ANSWER
40           */
41          this->y = y;
42      }
43
44      double getX() const
45      {
46          /*
47           * STUDENT ANSWER
48           */
49          return this->x;
50      }
51
52      double getY() const
53      {
54          /*
55           * STUDENT ANSWER
56           */
57          return this->y;
58      }
59
60      double distanceToPoint(const Point &pointA)
61      {
62
```

	Test	Input	Expected	Got	

✔	`Point pointO(0, 0);`
	`Circle A = Circle(pointO, 3);`
	`Circle B;`
	`B = A;`
	`cout << (B == A);`
			`1`	`1`	✔

✔	`Circle A;`
	`cin >> A;`
	`A.printCircle();`
		`2`
		`3.5`
		`2`
			`Center: {2.00, 3.50} and Radius 2.00`	`Center: {2.00, 3.50} and Radius 2.00`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 5

Chính xác

Điểm 1,00 của 1,00

In a game, we have class `Character` to store characters' data.
The class `Character` is declared as below:

```
class Character {
protected:
    int hp;
    int x;
    int y;
public:
    // Constructor: set the values of x and y and hp to 0
    Character();

    // Constructor: set the values of hp, x and y to each parameter
    Character(int hp, int x, int y);

    // Set and get hp
    int getHp();
    void setHp(int hp);

    // Set and get x
    int getX();
    void setX(int x);

    // Set and get y
    int getY();
    void setY(int y);

    // Get Manhattan distance to other character
    int getManhattanDistTo(Character* other);
};
```

Your task is to define the constructors and the methods of the class.

Note:
In this task, `iostream` library has been included, and `namespace std` is being used. No other libraries are allowed.

For example:

Test	Result
`Character ch1(100, 3, 6);` `cout << ch1.getHp() << " " << ch1.getX() << " " << ch1.getY();`	`100 3 6`

Answer: (penalty regime: 0 %)

Reset answer

```
    this->hp = hp;
}

int Character::getX()
{
    // STUDENT ANSWER
    return this->x;
}

void Character::setX(int x)
{
    // STUDENT ANSWER
    this->x = x;
}

int Character::getY()
{
    // STUDENT ANSWER
    return this->y;
}

void Character::setY(int y)
```

	Test	Expected	Got	
✔	`Character ch1(100, 3, 6);` `cout << ch1.getHp() << " " << ch1.getX() << " " << ch1.getY();`	`100 3 6`	`100 3 6`	✔
✔	`Character ch2;` `cout << ch2.getHp() << " " << ch2.getX() << " " << ch2.getY();`	`0 0 0`	`0 0 0`	✔

✔	`Character* ch31 = new Character(100, 1, 2);`
	`Character* ch32 = new Character(100, -3, 4);`
	`cout << ch31->getManhattanDistTo(ch32);`
	`delete ch31;`
	`delete ch32;`
		`6`	`6`	✔

✔	`Character ch4;`
	`ch4.setX(4);`
	`cout << ch4.getX();`
		`4`	`4`	✔

✔	`Character ch5;`
	`ch5.setY(5);`
	`cout << ch5.getY();`
		`5`	`5`	✔

✔	`Character ch6;`
	`ch6.setHp(6);`
	`cout << ch6.getHp();`
		`6`	`6`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 6

Chính xác

Điểm 1,00 của 1,00

In a game, we have class `Character` to store characters' data.
The class `Character` is declared as below:

```
class Character {
protected:
    int hp;
    int x;
    int y;
public:
    Character();
    Character(int hp, int x, int y);
    int getHp();
    void setHp(int hp);
    int getX();
    void setX(int x);
    int getY();
    void setY(int y);
    int getManhattanDistTo(Character* other);

    // Operator =: copy all data from Character other
    void operator=(const Character& other);

    // Operator <: Character a < Character b when a's hp is less than or equal b's hp
    bool operator<(const Character& other);

    // Operator () with zero parameters: print data of the instance with format: hp-x-y
    void operator()();
};
```

Your task is to overload these following operators: `=`, `<` and `()`. Their functions are described above.

Note:

In this task, `iostream` library has been included, and `namespace std` is being used. No other libraries are allowed.

For example:

Test	Result
`Character ch1(100, 3, 6);` `ch1();`	`100-3-6`

Answer: (penalty regime: 0 %)

Reset answer

```
{
    // STUDENT ANSWER
    Character tempt = other;
    this->setHp(tempt.getHp());
    this->setX(tempt.getX());
    this->setY(tempt.getY());
}

// Character a < Character b when a's hp is less than or equal b's hp
bool Character::operator<(const Character &other)
{
    // STUDENT ANSWER
    Character tempt = other;
    return this->getHp() <= tempt.getHp();
}

// Print data of the instance with format: hp-x-y
void Character::operator() ()
{
    // STUDENT ANSWER
    cout << this->getHp() << "-" << this->getX() << "-" << this->getY();
}
```

	Test	Expected	Got	
✔	`Character ch1(100, 3, 6);` `ch1();`	`100-3-6`	`100-3-6`	✔

✔	`Character ch21(10, 20, 30);`
	`Character ch22(5, 5, 6);`
	`cout << ((ch21 < ch22) ? "true" : "false");`
		`false`	`false`	✔

✔	`Character ch31;`
	`Character ch32;`
	`cout << ((ch31 < ch32) ? "true" : "false");`
		`true`	`true`	✔

✔	`Character ch4;`
	`ch4();`
	`cout << "\n";`
	`ch4 = Character(5, 10, 20);`
	`ch4();`
		`0-0-0`
		`5-10-20`
			`0-0-0`
			`5-10-20`
				✔

✔	`Character(3, 4, 5)();` `cout << ((Character(3, 4, 5) < Character(3, 4, 5)) ? "true" : "false");`	`3-4-5true`	`3-4-5true`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 7

Chính xác

Điểm 1,00 của 1,00

In a game, we have class `Character` to store characters' data.
The class `Character` is declared as below:

```
class Character {
private:
    int x;
    int y;
protected:
    int hp;
public:
    Character();
    Character(int hp, int x, int y);
    int getHp();
    void setHp(int hp);
    int getX();
    void setX(int x);
    int getY();
    void setY(int y);
    int getManhattanDistTo(Character* other);
    void operator()();
};
```

Your task is to define a new class Player which is a derived class of class Character. The requirements of the new class are listed below:

- Methods of base class `Character` cannot be accessed outside `Player` class using `Player` instances
  Example: `Player pl; pl.setX();` will raise errors when compiled.
- Player class has these methods and constructors:
  - Constructor `Player()`: acts just like `Character()`
  - Constructor `Player(int hp, int x, int y)`: acts just like `Character(hp, x, y)`
  - Method `void printPlayerData()`: prints data of the instance with format: `hp-x-y`
  - Method `void moveTo(int x, int y)`: sets the values of `x`, `y` to new values

- The mentioned constructors and methods can be accessed outside `Player` class.

Note:
In this task, `iostream` library has been included, and `namespace std` is being used. No other libraries are allowed.

For example:

Test	Result
`Player pl1(100, 3, 6);` `pl1.printPlayerData();`	`100-3-6`

Answer: (penalty regime: 0 %)

Reset answer

```
    Player()
    {
        hp = 0;
        x = 0;
        y = 0;
    }
    Player(int hp, int x, int y)
    {
        this->hp = hp;
        this->x = x;
        this->y = y;
    }
    void printPlayerData()
    {
        cout << this->hp << "-" << this->x << "-" << this->y;
    }
    void moveTo(int x, int y)
    {
        this->x = x;
        this->y = y;
    }
};
```

	Test	Expected	Got	
✔	`Player pl1(100, 3, 6);` `pl1.printPlayerData();`	`100-3-6`	`100-3-6`	✔
✔	`Player pl2;` `pl2.printPlayerData();`	`0-0-0`	`0-0-0`	✔

✔	`Player pl3(300, 1, 2);`
	`pl3.moveTo(3, 4);`
	`pl3.printPlayerData();`
		`300-3-4`	`300-3-4`	✔

✔	`Player pl4(300, 1, 2);`
	`const bool condition = (is_unambiguous_public_base_of<Character>(&pl4) == nullptr &&`
	`is_base_of<Character, Player>::value == true);`
	`assert(condition);`
				✔

✔	`Player pl5(300, 1, 2);`
	`pl5.moveTo(9, 7);`
	`pl5.printPlayerData();`
		`300-9-7`	`300-9-7`	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 8

Chính xác

Điểm 1,00 của 1,00

Hoang is a K19 student studying at Bach Khoa University. He plans to write a book management software for the library. In the class design, Hoang has designed the class Book as follows:

```
class Book
{
private:
    char* title;
    char* authors;
    int publishingYear;
public:
   // some method
}
```

Your task in this exercise is to implement functions marked with /* * STUDENT ANSWER */.

Note: For exercises in Week 2, we have #include <bits/stdc++.h> and using namespace std;

For example:

Test	Result

`Book book1("Giai tich 1","Nguyen Dinh Huy",2000);`
`book1.printBook();`
	`Giai tich 1`
	`Nguyen Dinh Huy`
	`2000`

`Book book1("Giai tich 1","Nguyen Dinh Huy",2000);`
`Book book2 = book1;`
`book2.printBook();`
	`Giai tich 1`
	`Nguyen Dinh Huy`
	`2000`

Answer: (penalty regime: 0 %)

Reset answer

```
41      void setTitle(const char *title)
42      {
43          /*
44           * STUDENT ANSWER
45           */
46          char* t=(char*) title;
47          this->title = t;
48      }
49
50      void setAuthors(const char *authors)
51      {
52          /*
53           * STUDENT ANSWER
54           */
55          char* a = (char*) authors;
56          this->authors = a;
57      }
58
59      void setPublishingYear(int publishingYear)
60      {
61          /*
62           * STUDENT ANSWER
63           */
64          this->publishingYear = publishingYear;
65      }
66
67      char *getTitle() const
68      {
69          /*
70           * STUDENT ANSWER
71           */
72          return this->title;
73      }
74
75      char *getAuthors() const
76      {
77          /*
78           * STUDENT ANSWER
79           */
```

```
        return this->authors;
    }

    int getPublishingYear() const
    {
        /*
         * STUDENT ANSWER
         */
        return this->publishingYear;
    }
```

	Test	Expected	Got	

✔	`Book book1("Giai tich 1","Nguyen Dinh Huy",2000);`
	`book1.printBook();`
		`Giai tich 1`
		`Nguyen Dinh Huy`
		`2000`
			`Giai tich 1`
			`Nguyen Dinh Huy`
			`2000`
				✔

✔	`Book book1("Giai tich 1","Nguyen Dinh Huy",2000);`
	`Book book2 = book1;`
	`book2.printBook();`
		`Giai tich 1`
		`Nguyen Dinh Huy`
		`2000`
			`Giai tich 1`
			`Nguyen Dinh Huy`
			`2000`
				✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 9

Chính xác

Điểm 1,00 của 1,00

In this exercise, you can use implemented functions in *previous question* (if needed) and implement these following functions.

```
friend bool checkAuthor(Book book, char* author){}
```

In the authors attribute, it is possible to have more than one author writing a book together. So authors will have the following format: "author1, author2, ..., authorN"

The function returns true if the author is on the book's authors list, otherwise it returns false

Note: Both first and last name must match. If only a partial match, the function still returns false

For example:

Test	Result
`Book book1("Giai tich 1","Nguyen Dinh Huy, Nguyen Thi Xuan Anh",2000);` `cout << checkAuthor(book1,"Nguyen Dinh Huy");`	`1`
`Book book1("Giai tich 1","Nguyen Dinh Huy, Nguyen Thi Xuan Anh",2000);` `cout << checkAuthor(book1,"Nguyen Thi Xuan");`	`0`

Answer: (penalty regime: 0 %)

Reset answer

```
46          char *t = (char *)title;
47          this->title = t;
48      }
49
50      void setAuthors(const char *authors)
51      {
52          /*
53           * STUDENT ANSWER
```

```
         */
        char *a = (char *)authors;
        this->authors = a;
    }

    void setPublishingYear(int publishingYear)
    {
        /*
         * STUDENT ANSWER
         */
        this->publishingYear = publishingYear;
    }

    char *getTitle() const
    {
        /*
         * STUDENT ANSWER
         */
        return this->title;
    }

    char *getAuthors() const
    {
        /*
         * STUDENT ANSWER
         */
        return this->authors;
    }

```

	Test	Expected	Got	
✔	`Book book1("Giai tich 1","Nguyen Dinh Huy, Nguyen Thi Xuan Anh",2000);` `cout << checkAuthor(book1,"Nguyen Dinh Huy");`	1	1	✔
✔	`Book book1("Giai tich 1","Nguyen Dinh Huy, Nguyen Thi Xuan Anh",2000);` `cout << checkAuthor(book1,"Nguyen Thi Xuan");`	0	0	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 10

Chính xác

Điểm 1,00 của 1,00

In this exercise, you will implement function **printBook(const Book book)** in **class Printer** to print information of the book. See example for output format (no spaces at the end of each line and no empty lines at the end).

Note: In the authors attribute, it is possible to have more than one author writing a book together. So authors will have the following format: "author1, author2, ..., authorN"

For example:

Test	Result

`Book book1("Giai tich 1", "Nguyen Dinh Huy, Nguyen Thi Xuan Anh", 2000);`
`Printer::printBook(book1);`
	`Giai tich 1`
	`Nguyen Dinh Huy`
	`Nguyen Thi Xuan Anh`
	`2000`

`Book book1("Introduction to Algorithms", "Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein", 1990);`
`Printer::printBook(book1);`
	`Introduction to Algorithms`
	`Thomas H. Cormen`
	`Charles E. Leiserson`
	`Ronald L. Rivest`
	`Clifford Stein`
	`1990`

Answer: (penalty regime: 0 %)

Reset answer

```
1  class Book
2  {
3  private:
```

```
    char *title;
    char *authors;
    int publishingYear;

public:
    Book()
    {
        /*
         * STUDENT ANSWER
         * TODO: set zero publishingYear and null pointer
         */
        title = nullptr;
        authors = nullptr;
        publishingYear = 0;
    }

    Book(const char *title, const char *authors, int publishingYear)
    {
        /*
         * STUDENT ANSWER
         */
        this->setTitle(title);
        this->setAuthors(authors);
        this->setPublishingYear(publishingYear);
    }

    Book(const Book &book)
    {
        /*
         * STUDENT ANSWER
         * TODO: deep copy constructor
         */
        this->setTitle(book.getTitle());

```

	Test	Expected	Got	

✔	`Book book1("Giai tich 1", "Nguyen Dinh Huy, Nguyen Thi Xuan Anh", 2000);`
	`Printer::printBook(book1);`
		`Giai tich 1`
		`Nguyen Dinh Huy`
		`Nguyen Thi Xuan Anh`
		`2000`
			`Giai tich 1`
			`Nguyen Dinh Huy`
			`Nguyen Thi Xuan Anh`
			`2000`
				✔

✔	`Book book1("Introduction to Algorithms", "Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein", 1990);`
	`Printer::printBook(book1);`
		`Introduction to Algorithms`
		`Thomas H. Cormen`
		`Charles E. Leiserson`
		`Ronald L. Rivest`
		`Clifford Stein`
		`1990`
			`Introduction to Algorithms`
			`Thomas H. Cormen`
			`Charles E. Leiserson`
			`Ronald L. Rivest`
			`Clifford Stein`
			`1990`
				✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 11

Chính xác

Điểm 1,00 của 1,00

1. In the toy store, all toy has a price. Car toy has a price and color, Puzzle toy has a price and size. We have to implement class CarToy and class PuzzleToy which inherit from class Toy.
2. class ToyBox has a pointer array to store a list of toys (up to 5 items including car and puzzle) and number of items in the box.

Your task is to implement two function addItem(...) in class ToyBox. If successfully added, the function returns the current number of toys in the box. If the box is full, return -1.

For example:

Test	Result

`CarToy car(20000,red);`
`PuzzleToy puzzle(30000,small);`
`car.printType();`
`puzzle.printType();`
	`This is a car toy`
	`This is a puzzle toy`

`CarToy car(20000,red);`
`PuzzleToy puzzle(30000,small);`
 	
`ToyBox box;`
`box.addItem(car);`
`box.addItem(puzzle);`
`box.printBox();`
	`This is a car toy`
	`This is a puzzle toy`

`Toy* toy = new CarToy(30000,red);` `toy->printType();`	`This is a car toy`

Answer: (penalty regime: 0 %)

Reset answer

```
    {
        for (int i = 0; i < numberOfItems; ++i)
        {
            delete toyBox[i];
        }
    }

    int addItem(const CarToy &carToy)
    {
        /*
         * STUDENT ANSWER
         * TODO: function add a new Car toy to the box.
                If successfully added, the function returns the current number
                If the box is full, return -1.
         */
        if (numberOfItems + 1 > 5)
            return -1;
        else
        {
            toyBox[numberOfItems] = new CarToy(carToy);
            ++numberOfItems;
            return numberOfItems;
        }
    }

    int addItem(const PuzzleToy &puzzleToy)
    {
        /*
         * STUDENT ANSWER
         * TODO: function add a new Puzzle toy to the box.
                If successfully added, the function returns the current number
                If the box is full, return -1.
         */
        if (numberOfItems + 1 > 5)
            return -1;
        else
        {
            toyBox[numberOfItems] = new PuzzleToy(puzzleToy);
            ++numberOfItems;
            return numberOfItems;
```

```
137         }
138     }
139
140     void printBox()
141     {
142         for (int i = 0; i < numberOfItems; i++)
143             toyBox[i]->printType();
144     }
145 };
```

	Test	Expected	Got	

✔	CarToy car(20000,red);
	PuzzleToy puzzle(30000,small);
	car.printType();
	puzzle.printType();
		This is a car toy
		This is a puzzle toy
			This is a car toy
			This is a puzzle toy
				✔

✔	CarToy car(20000,red);
	PuzzleToy puzzle(30000,small);
	
	ToyBox box;
	box.addItem(car);
	box.addItem(puzzle);
	box.printBox();
		This is a car toy
		This is a puzzle toy
			This is a car toy
			This is a puzzle toy
				✔

✔	Toy* toy = new CarToy(30000,red); toy->printType();	This is a car toy	This is a car toy	✔

Passed all tests! ✔

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

BÁCH KHOA E-LEARNING

WEBSITE

HCMUT

MyBK

BKSI

LIÊN HỆ

268 Lý Thường Kiệt, P.14, Q.10, TP.HCM

(028) 38 651 670 - (028) 38 647 256 (Ext: 5258, 5234)

elearning@hcmut.edu.vn