You are working on problem set: Project 5 (Pause)

polymorphismMysteryHamburger

Language/Type: C++ inheritance polymorphism

Consider the following classes; assume that each is defined in its own file.

```
class Hamburger : public Bacon {
public:
    virtual void m2() {
        cout << "H 2" << endl;
        Bacon::m2();
    }

    virtual void m4() {
        cout << "H 4" << endl;
    }
};

class Mayo : public Hamburger {
public:
    virtual void m3() {
        cout << "M 3" << endl;
        m1();
    }

    virtual void m4() {
        cout << "M 4" << endl;
    }
};

class Lettuce {
public:
    virtual void m1() {
        cout << "L 1" << endl;
```

```
        m2();
    }

    virtual void m2() {
        cout << "L 2" << endl;
    }
};

class Bacon : public Lettuce {
public:
    virtual void m1() {
        Lettuce::m1();
        cout << "B 1" << endl;
    }

    virtual void m3() {
        cout << "B 3" << endl;
    }
};
```

Now assume that the following variables are defined:

```
Lettuce* var1 = new Bacon();
Bacon* var2 = new Mayo();
Lettuce* var3 = new Hamburger();
Bacon* var4 = new Hamburger();
Lettuce* var5 = new Lettuce();
```

In the table below, indicate in the right-hand column the output produced by the statement in the left-hand column. If the statement produces more than one line of output, indicate the line breaks with slashes as in "x / y / z" to indicate three lines of output with "x" followed by "y" followed by "z". If the statement does not compile, write "COMPILER ERROR". If a statement would crash at runtime or cause unpredictable behavior, write "CRASH".

.
.
.

`var1->m1();`	`L 1 / L 2 / B 1`
`var1->m2();`	`L 2`
`var1->m3();`	`COMPILER ERROR`
`var2->m1();`	`L 1 / H 2 / L 2 / B 1`
`var2->m2();`	`H 2 / L 2`

var2->m3();	M 3 / L 1 / H 2 / L 2 / B
var2->m4();	COMPILER ERROR
var3->m1();	L 1 / H 2 / L 2 / B 1
var3->m2();	H 2 / L 2
var4->m2();	H 2 / L 2
var4->m3();	B 3
var4->m4();	COMPILER ERROR
((Bacon*) var1)->m1();	L 1 / L 2 / B 1
((Bacon*) var1)->m3();	B 3
((Mayo*) var5)->m3();	CRASH
((Lettuce*) var4)->m3();	COMPILER ERROR
((Hamburger*)var2)->m4();	M 4
((Mayo*) var2)->m4();	M 4

Submit

You passed 18 of 18 tests.

#	question	your answer	result
1	var1->m1();	L 1 / L 2 / B 1	pass
2	var1->m2();	L 2	pass
3	var1->m3();	COMPILER ERROR	pass
4	var2->m1();	L 1 / H 2 / L 2 / B 1	pass
5	var2->m2();	H 2 / L 2	pass
6	var2->m3();	M 3 / L 1 / H 2 / L 2 / B 1	pass
7	var2->m4();	COMPILER ERROR	pass
8	var3->m1();	L 1 / H 2 / L 2 / B 1	pass
9	var3->m2();	H 2 / L 2	pass
10	var4->m2();	H 2 / L 2	pass
11	var4->m3();	B 3	pass
12	var4->m4();	COMPILER ERROR	pass
13	((Bacon*) var1)->m1();	L 1 / L 2 / B 1	pass
14	((Bacon*) var1)->m3();	B 3	pass
15	((Mayo*) var5)->m3();	CRASH	pass
16	((Lettuce*) var4)->m3();	COMPILER ERROR	pass
17	((Hamburger*)var2)->m4();	M 4	pass
18	((Mayo*) var2)->m4();	M 4	pass

Need help?

Stuck on an exercise? Contact your TA or instructor .

If something seems wrong with our site, please contact us.