CLA_CSE-I_12-11-2022

justin_t@srmap.edu.in Switch account

Your email will be recorded when you submit this form

* Required

What will be the output of the following statement? *

cout << (9+1)/double(int(3.45));

- ○ 3
- ○ Compilation Error.
- ○ 0
- ○ 3.33333

Which among these is not true about static data members in C++? *

- ○ They need to be redeclared using class name using scope resolution outside the class.
- ○ They are available even before the creation of the objects of any class.
- ○ They need to be initialized to zero during declaration.
- ○ The value of the static variables is the same for all the global variables in C++.

Predict the output of the following program. *

```
#include<iostream>
using namespace std;

int g(int x)
{
    int tmp = x + 1;
    return x * tmp;
}
int f(int x, int y)
{
    x += y;
    return g(x);
}
int main() {
    int x = 3;
    cout << g(x) <<" ";
    cout << f(x, 1) << " ";
    cout << f(x, 2) << " ";
}
```

- ○ 12 20 30
- ○ 10 20 30
- ○ 20 12 30
- ○ 30 20 12

Which among the following statements is incorrect? *

- Protected members of the base class can be accessed directly in the derived class no matter what mode of inheritance we are using.
- Private members of the base class can't be accessed in the derived class irrespective of the mode of inheritance.
- Public members of the base class can't be accessed in the main function through derived class object, if the mode of inheritance is protected.
- Public members of the base class can't be redeclared as public in the derived class if the mode of inheritance is private.

What will be the output of the following code? *

```
#include <iostream>
using namespace std;

void f(int x);
void f(int &x);
int main()
{
    int a=10;
    f(a);
    return 0;
}
void f(int x)
{
    cout << "In f(int)\n";
}
void f(int &x)
{
    cout << "In f(int &)\n";
}
```

- ◯ In f(int)
- ◯ In f(int &)
- ◯ Compilation Error
- ◯ Run Time Error

Which among these is the correct option to free the dynamically allocated memory for an array. (Let the pointer name is ptr) *

- delete ptr[];
- delete[] ptr;
- delete *ptr;
- delete ptr;

In case of binary operator overloading with member function, which of following statement should be taken into consideration? *

- a. Right hand operand must be object.
- b. Left hand operand must be object.
- c. Both the operands must be objects.
- d. All of these should be considered.

Which among the following is incorrect about static member? *

- It is initialized before any object of that class is created.
- Its lifetime is within the entire program.
- Static members can be accessed directly using class name.
- Global definition of static member is optional.

What will be the output of the following program? *

```
#include<iostream>
using namespace std;
class Test
{
    public:
        int a;
        Test()
        {
            cout<<"Constructor Executed."<<endl;
        }
        ~Test()
        {
            cout<<"Destructor Executed."<<endl;
        }
};
class Test2
{
    int b;
    public:
        void setB(int x, int y)
        {
            Test obj;
            obj.a=x;
            b=y;
        }
};
int main()
{
    Test2 obj;
    obj.setB(10,20);
    Test obj1,obj2;
    return 0;
}
```

Constructor Executed.
Destructor Executed.
Constructor Executed.
Destructor Executed.
Constructor Executed.
Destructor Executed.

Option 1

Constructor Executed.
Constructor Executed.
Constructor Executed.
Destructor Executed.
Destructor Executed.
Destructor Executed.

Option 2

Constructor Executed.
Destructor Executed.
Constructor Executed.
Constructor Executed.
Destructor Executed.
Destructor Executed.

Option 3

None

How many times the cout statement will be executed for the following code fragment? *

```
for( i=0;i<5;i++);
    cout<<i;
```

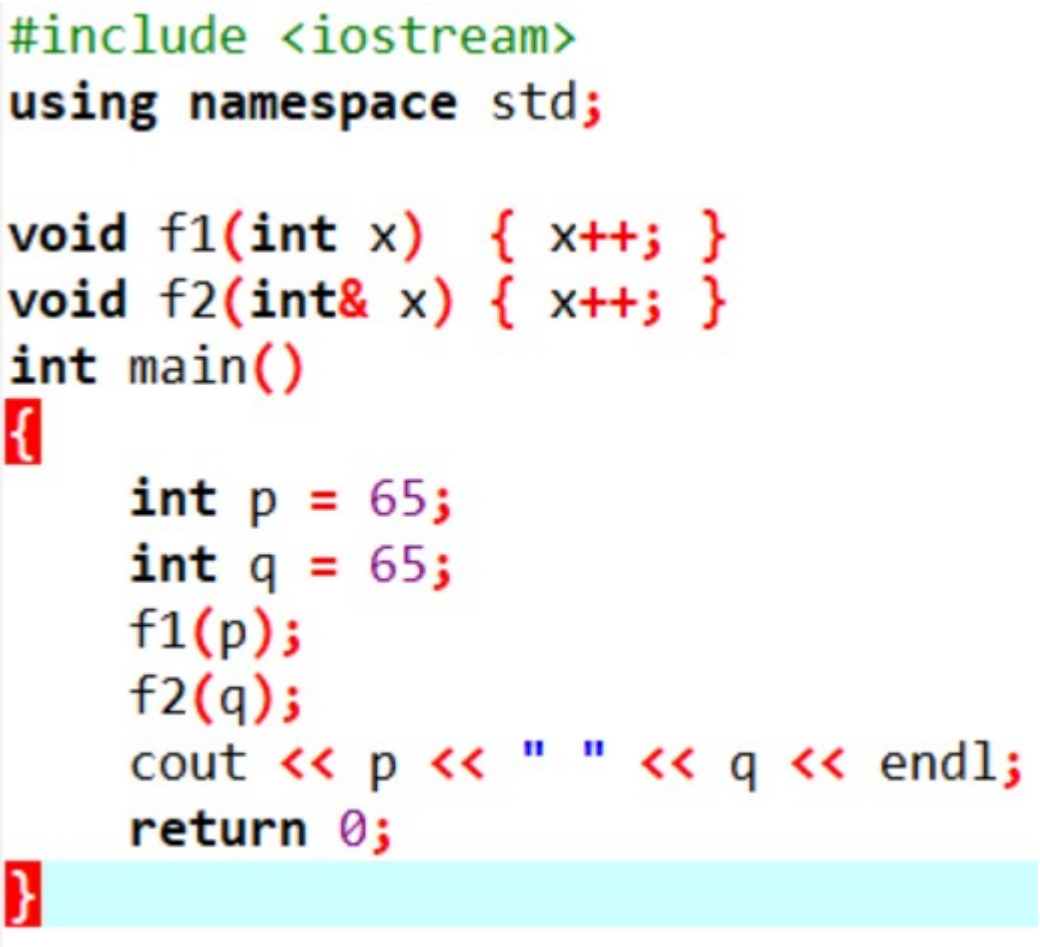
0

1

5

4

What will be the output of the following code? *



```
#include <iostream>
using namespace std;

void f1(int x)  { x++; }
void f2(int& x) { x++; }
int main()
{
    int p = 65;
    int q = 65;
    f1(p);
    f2(q);
    cout << p << " " << q << endl;
    return 0;
}
```

- ○ 65 66
- ○ 66 66
- ○ 66 65
- ○ 65 65

Pick the correct print statement for the following piece of code. *

```
#include <iostream>

void f (int& p1, int& p2) {
   p1 = p2 + 7 * 2;
}

int main (void) {
   int x, y;

   x = 2;
   f (x, x);

   y = x;
   f (y, y);

   cout << x << endl;
   return 0;
}
```

- ○ 18
- ○ 16
- ○ 30
- ○ 2

Which among the following statements is true? *

- Ambiguity in function call occurs when we have two copies of the same functions available in multilevel inheritance.
- Ambiguity in function call occurs when we have two copies of the same functions available in multiple inheritance.
- Ambiguity in function call occurs when two function with exact same signature defined in both base and derived classes.
- Ambiguity in function call occurs when we have define two functions with the same name in both derived and base class.

Overloading unary operators (through member function) will require ___ number of arguments. *

- 1
- 2
- 0
- Not Fixed

cout is a: *

- An object
- A library function
- A system file
- A class

What will be the state of a and b after the following two function calls? *

bool a=func(5,7);
bool b=func(0,4);

```
bool func(int l,int h)
{
    if(l>h)
        return false;
    if(l==h)
        return true;
    else
        return func(l+2,h+1);
}
```

- ○ a is false, b is false
- ○ a is false b is true
- ○ both a and b are true
- ○ a is true and b is false

What will be the output of the following code? *

```
#include <iostream>
using namespace std;
void swap(int & a, int b)
{
    int temp = a;
    a = b;
    b = temp;
}
int main()
{
    int a = 4;
    int b = 2;
    int c = 5;
    swap(a,b);
    swap(b,c);
    cout << a << " " << b << " " << c;
}
```

- 5 2 5
- 5 5 2
- 2 5 2
- 2 5 5

```
#include <stdio.h>      *
int main(void)
{
    int new = 5;
    printf("%d", new);
}
```

- a) Error in C and successful execution in C++
- b) Error in both C and C++
- c) Error in C++ and successful execution in C
- d) A successful run in both C and C++

What will be the output of the following? *

```
#include<iostream>
using namespace std;
class Test
{
    int a;
    public:
        Test(int x)
        {
            a=x;
        }
        void show()
        {
            cout<<a<<endl;
        }
};

int main()
{
    Test obj=15;
    obj.show();
    return 0;
}
```

- ○ Compilation Error
- ○ 15
- ○ Run Time Error
- ○ None

Which of the following statements is incorrect? *

- ○ The default constructor of the base class is called automatically called when an object of the derived class is created.
- ○ We need at least one arguments to call the parameterized constructor (having one argument) of the base class through derived class.
- ○ We always have to define default constructor if we are defining a parameterized constructor in the base class during inheritance.
- ○ We can leave blank of a parameterized constructor in derived class.

Submit Clear form