

[Quiz](#) >

## Review answers



Complete date: A moment ago

Question 1: Which statement is true about operator overloading?

- ☐ Operator overloading enable us to define a special member function that is called when you use an operator on an instance of the class.
- ☐ The overload operator is a special operator to enable function overloading.
- ☒ With operator overloading we can give a different meaning to existing operators e.g. when using the + operator on two doubles.
- ☐ Operator functions must be member functions.

Question 2: What statement is true about the following code?

C++:

```
1 namespace A::B
2 {
3     class MyClass
4     {
5     };
6 }
```

- ☒ This code defines a class in namespace *B* that is nested in namespace *A*.
- ☐ This code defines a class in namespace *A* that is nested in namespace *B*.
- ☐ This code does not compile.
- ☐ This code defines a class in the single namespace called *A::B*.

Question 3: Which statement is false about friends?

- ☐ Friend functions can access the private members of the class they are friend of.
- ☐ A class cannot access the private members of its friend classes.
- ☐ Friend functions can access the 'this' pointer

Friends functions can access the `this` pointer.

☒ Friends violate the information hiding principle of object-oriented programming.

Question 4: Which statement is false about the following code assuming class *MyClass* has a member function called *F()*?

C++:

```
1 | MyClass* mc=new MyClass;
```

- ☒ The code is wrong because you need to add round brackets when dynamically creating an object: `new MyClass()`
- ☐ The code creates a single *MyClass* object on the heap using the default constructor.
- ☐ Function *F()* can be called as follows: `(*mc).F()`;
- ☐ Function *F()* can be called as follows: `mc->F()`;

Question 5: Which statement is false about namespaces?

- ☐ Namespaces can be nested.
- ☒ A namespace must be compiled in its own `.lib` file.
- ☐ Multiple namespace blocks with the same name are possible.
- ☐ Namespaces can prevent name collisions or be used to group functionality in logical blocks.

Question 6: What statement is true about the following code?

C++:

```
1 | namespace
2 | {
3 |     int x=20;
4 | }
```

- ☐ This does not compile because the namespace has no name.
- ☐ Variable *x* is inaccessible because the namespace has no name.
- ☐ The variable *x* is a global variable only accessible in the current compilation unit and not as external global variable in another compilation unit.
- ☒ This code defines a local variable *x* in its own scope.

Question 7: Which statement is true about the following code?

Question 7: Which statement is true about the following code:

C++:



```
1 | delete[] x;
```

- ☐ The code is wrong because the `[]` are missing the size to delete.
- ☐ The code is wrong because the `[]` are not supported with delete.
- ☒ It deallocates an array pointed by variable x.
- ☐ It deallocates the first element of an array pointed by variable x.

Question 8: What is the output of the following program?

C++:



```
1 | int x=10;
2 |
3 | namespace
4 | {
5 |     int x=20;
6 | }
7 |
8 | int main(int x, char* y[])
9 | {
10 |     {
11 |         int x=30;
12 |         std::cout<<:x<<std::endl;
13 |     }
14 |     return 0;
15 | }
```

- ☒ 10
- ☐ 20
- ☐ 30
- ☐ The number of arguments passed to the program.

Question 9: Which of the options below is the best description of a canonical header file?

- ☐ A header file with minimal a constructor, destructor, assignment operator and equal compare operator.
- ☐ A header file with minimal a default constructor, copy constructor, destructor and equal compare operator.
- ☐ A header file with minimal a constructor, destructor, equal compare operator and not equal compare operator.
- ☒ A header file with minimal a default constructor, copy constructor, destructor and assignment operator.

Question 10: Which of the following options declares and allocates an array of *int* pointers?

- ☐ `int*[] array=new int*[size];`
- ☐ `int[] array=new int*[size];`
- ☐ `int** array=new int*[size];`
- ☒ `int* array=new int*[size];`

Score: 5 (50.00%)

Pass/Fail: Failed

[Contact us](#) [Advertise](#) [Terms and rules](#) [Privacy policy](#) [Help](#) [Home](#) 

© QUANTNET INC