

Quiz 3 Smart Pointers

© Datasim Education BV 2018

1. What can be said about `std::auto_ptr<>` (true/false)?
 - a) It is supported in C++11 and deprecated in C++14.
 - b) It supports *move semantics*.
 - c) It only supports the creation of single objects.
 - d) Loss of ownership is an issue.
2. What are the main features of *smart pointers* in C++ in general?
 - a) They employ garbage collection to manage memory allocation and deallocation.
 - b) They support both *exclusive* and *shared ownership* of objects and memory.
 - c) They know whether they are the last owner of a resource.
 - d) Objects are cleaned up as soon as they are no longer needed.
3. What is a *dangling pointer*?
 - a) A pointer that has been deleted.
 - b) A pointer that points to memory that no longer holds the object that the pointer is supposed to point to.
 - c) When a null pointer is deleted.
 - d) It occurs when a `std::auto_ptr` is copied.
4. Which entities are in a shared pointer's *control block*?
 - a) Custom deleter.
 - b) Scoped count.
 - c) Reference count.
 - d) Weak count.
5. Which of the following statements are true?
 - a) A weak pointer has a reference count.
 - b) A weak pointer is used to break *circular references* in shared pointers.
 - c) Smart pointers are thread-safe.
 - d) C++11 unique pointers are similar to Boost scoped pointers.
6. Consider the following code:

```
// Simple code to show use count
std::cout << "\nSome stuff with smart pointers\n";
using SP = std::shared_ptr<double>;
using WP = std::weak_ptr<double>;

SP a(new double);
*a = 3.1415;
std::cout << "Use count: " << a.use_count() << '\n'; // A

SP a2(a);
std::cout << "Use count: " << a2.use_count() << '\n'; // B
```

```
WP w(a2);
std::cout << "Use count: " << w.use_count() << '\n'; // C
std::cout << "Expired: " << std::boolalpha << w.expired(); // D

w.reset();
std::cout << "Use count: " << w.use_count() << '\n'; // E

a2.reset();
std::cout << "Use count: " << a2.use_count() << '\n'; // F

std::cout << "Use count: " << a.use_count() << '\n'; // G
```

What is the output from A to G?

- a) {1,2,2,true,0,0,1}.
- b) {1,2,2,false,0,0,1}.
- c) {1,2,2,false,0,0,0}.
- d) {1,2,2,true,0,0,0}.