Question 3 - Debugging (20)

The code below contains one syntactic/logical error under C++17 standard (errors caught by a compiler or generated crashes/memory leaks/undefined behaviour at runtime).

#include <iostream>

#define SQUARE(x) x \* x

int main() {

int a = 5;

int result = SQUARE(a + 1);

std::cout << "The square of (a + 1) is: " << result << std::endl;

return 0;

}

#include <iostream>

class Base {

public:

virtual void show() { std::cout << "Base class" << std::endl; }

};

class Derived : public Base {

public:

void show() override { std::cout << "Derived class" << std::endl; }

};

int main() {

Base\* b = new Derived();

delete b;

return 0;

}

The above code runs but has a memory management issue. Identify the problem and fix it.

#include <iostream>

class A {

public:

void show() { std::cout << "Class A" << std::endl; }

};

class B {

public:

void show() { std::cout << "Class B" << std::endl; }

};

class C : public A, public B {};

int main() {

C obj;

obj.show();

return 0;

}

The above code will not run due to an error. Identify the problem and fix it.

Question 2: walkthrough (25 marks)

What is the output of the following program? (10 marks)

1. #include <iostream>
2. int main() {
3. int arr[3] = {1, 2, 3};
4. int\* ptr1 = arr;
5. int (\*ptr2)[3] = &arr;
6. std::cout << \*(ptr1) << " ";
7. std::cout << \*(ptr1 + 1) << " ";
8. std::cout << \*(ptr1 + 2) << std::endl;
9. std::cout << (\*ptr2)[0] << " ";
10. std::cout << (\*ptr2)[1] << " ";
11. std::cout << (\*ptr2)[2] << std::endl;
12. return 0;

}

Question 4 - Programming (43)

Write a C++ program that defines a Functons-Like Macros called MIN, which takes two arguments and returns the smaller of the two. Demonstrate its usage with two integers and two floating-point numbers.

Explain how virtual inheritance is utilized in this code to prevent the "diamond problem," and describe the role of the display function call in the main function.

#include <iostream>

class A {

public:

A(int a, int b ){}

void display() { std::cout << "Display from A" << std::endl; }

};

class B : virtual public A {};

class C : virtual public A {};

class D : public B, public C {};

int main() {

D obj;

obj.display();

return 0;

}

April 7:

Question 2: walkthrough (25 marks)

What is the output of the following program? (10 marks)

#include <iostream>

#include <list>

#include <string>

int main() {

std::list<std::string> cities;

cities.push\_back("Toronto");

cities.push\_back("Vancouver");

cities.push\_back("Montreal");

std::list<std::string>::iterator it = cities.begin();

++it;

cities.insert(it, "Calgary");

for (const auto& city : cities) {

std::cout << city << std::endl;

}

return 0;

}

#include <iostream>

#include <vector>

#include <deque>

int main() {

int size = 6;

int\* arr = new int[size];

for (int i = 0; i < size; ++i) {

arr[i] = i \* 3;

}

for (int i = 0; i < size; ++i) {

std::cout << arr[i] << " ";

}

std::cout << std::endl;

std::deque<int> dq(arr, arr + size);

for (int val : dq) {

std::cout << val << " ";

}

delete[] arr;

return 0;

}

#include <iostream>

#include <list>

#include <algorithm>

int main() {

std::list<std::string> colors = {"Red", "Green", "Blue", "Yellow"};

/// Which stl Algorithm do you have to use here to get the output below (don’t use for\_each)

for (const auto& color : colors) {

std::cout << color << " ";

}

return 0;

}

Output is:

Yellow Blue Green Red

#include <iostream>

#include <vector>

#include <algorithm>

int main() {

std::vector<int> numbers = {10, 20, 30, 40, 50};

/// Which stl Algorithm do you have to use here to get the output below (don’t use for\_each)

for (const auto& num : numbers) {

std::cout << num << " ";

}

return 0;

}

Output: 20 30 40 50 60

#include <iostream>

#include <vector>

#include <algorithm>

int main() {

std::vector<int> numbers = {10, 20, 30, 40, 50, 60};

std::vector<int> modifiedNumbers;

/// Which stl Algorithm do you have to use here to get the output below (don’t use for\_each)

std::vector<int> filteredNumbers;

/// Which stl Algorithm do you have to use here to get the output below (don’t use for\_each)

std::cout << "Modified vector: ";

for (const auto& num : modifiedNumbers) {

std::cout << num << " ";

}

std::cout << std::endl;

std::cout << "Filtered vector: ";

for (const auto& num : filteredNumbers) {

std::cout << num << " ";

}

std::cout << std::endl;

return 0;

}

/// output

Modified vector: 15 25 35 45 55 65

Filtered vector: 35 45 55 65

Programming Questions:

**Using STL Algorithm to Remove Even Numbers**

Write a C++ program that initializes a std::vector<int> with values from 1 to 10 and use any stl algorithm to remove all even numbers from the vector. Then outputs the modified vector.

April 10

1. Describe the concept of ownership in the context of std::unique\_ptr and std::shared\_ptr. What happens when you try to copy a std::unique\_ptr?
2. Explain the difference between #include <file> and #include "file" in C++. When would you use each?
3. Given an array int arr[] = {5, 10, 15, 20};, write a pointer expression to access the third element and increment its value.
4. What is the difference between std::vector and std::list in terms of memory allocation and performance for insertions and deletions?
5. What is a deadlock in multithreading, and how can it be avoided? Provide an example of a potential deadlock situation.
6. What is the role of the #undef directive in C++? Provide a scenario where it might be used.
7. What is the difference between \*(arr + i) and arr[i] in C++? Explain with an example.

Cout << \*(arr +i);

Cout << arr[i];

1. What is the result of the following C++ code, and why?

A computer screen with text

AI-generated content may be incorrect.

Locate one syntax error in this code:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <cstring>

#include <iomanip>

#include <memory>

using namespace std;

int main() {

unique\_ptr<int> ptr (new int);

cout << "int value: ";

cin >> \*ptr;

cout << \*ptr << endl;

unique\_ptr<int> ptr2 = ptr;

cout << \*ptr2 << endl;

return 0;

}

Locate one syntax error in this code:

#include <iostream>

#include <cstring>

using namespace std;

int main() {

int ints[]{ 1,3,4,5,6,7,8 };

for\_each(ints, ints + 5,

[](int p) {

cout << p << " ";

});

cout << endl;

}

What is the output of this code:

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <cstring>

#include <iomanip>

#include <memory>

using namespace std;

int main() {

shared\_ptr<int> a(new int);

\*a = 12;

cout << \*a << endl;

shared\_ptr<int> b(a);

shared\_ptr<int> c(b);

(\*c)++;

shared\_ptr<int> d = move(a);

if (a != nullptr)

cout << \*a << endl;

if (d != nullptr)

cout << \*d << endl;

cout << \*b << " "<<\*c << endl;

return 0;

}

1. Compare and contrast std::unique\_ptr and std::shared\_ptr in C++.
2. Describe the difference between a race condition and a deadlock in multithreading. Provide an example scenario for each.
3. Explain how a mutex can be used to prevent data races in a multithreaded application. Provide a code example.
4. Explain the purpose of the #define pre-processor directive in C++. Provide an example of its usage and explain how it can affect code readability and maintainability.
5. Describe how you can use pointers to iterate through an array in C++. Provide a code example.
6. What is the diamond problem in multiple inheritance, and how does C++ resolve it?
7. What is the output?

int main() {

     int a[3][5] = {

      {1,2,3,4,5},

      {6,7,8,9,10},

      {11,12,13,14,15}

};

int\* p = (int\*)a;

for (i = 0; i < 15; i++) {

      cout << \*(p + i) << " ";

   }

return 0;

}