

Student Number:

Name:

Write down answers in-between questions. Please answer using short sentences.

The back of each page can be used for practice, but DO NOT write down the answer on the back.

Be sure to write your student number and name on each page.

1 point

1. If a member function is defined inside a class definition, the compiler generates code for it as an inline function. ()
2. Using `std::list` will always produce faster performance than using `std::vector`. ()
3. The elements of `std::list` are stored in contiguous memory. ()
4. C++ structures can have member functions. ()
5. Only one constructor should exist in a class. ()
6. Class member functions must be specified as public. ()
7. If you don't set any access specifier for class members, they are considered private. ()
8. You can assign a value to a name of an array and use it like a pointer. ()
9. A dynamically allocated memory space can only be accessed through a pointer. ()
10. `std::vector` is a sequential access data structure. ()

4 points

11. Write down the expected output of the following program.

Student Number:

Name:

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

int main () {
    int a = 10;
    int b = a;
    int* p = &a;
    int& r = a;

    a = 20;
    b = 30;

    *p = 10;
    r = 40;

    p = &b;
    *p = 50;

    cout << a << endl;
    cout << b << endl;
    cout << *p << endl;
    cout << r << endl;

    return 0;
}
```

12. After running make once using the Makefile below, you modified the read.c file. What are newly created files when you run make again?

```
test : read.o write.o main.o
      gcc -o test read.o write.o main.o

main.o : io.h main.c
      gcc -c main.c

read.o : io.h read.c
      gcc -c read.c

write.o: io.h write.c
      gcc -c write.c
```

13. Write down the expected output of this program.

Student Number:

Name:

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

int main ()
{
    vector<int> vec(5);
    for(int i=0; i<vec.size(); ++i)
        vec[i] = 2*i;

    vector<int>::iterator it = vec.begin();
    *it = 8;
    *(it+2) = 9;

    for(int i=0; i<vec.size(); ++i)
        cout << vec[i] << " ";
    cout << endl;

    return 0;
}
```

14. Write down the expected output of this program.

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

int main ()
{
    string s = "0123456789";
    size_t pos = s.find("345");
    string s2 = s.substr(pos,5);
    cout << s2 << endl;

    return 0;
}
```

15. What is the expected output of the following program?

Student Number:

Name:

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

class A
{
public:
    A() { cout << "a1 "; }
    ~A() { cout << "a2 "; }
};

class B
{
public:
    B() { cout << "b1 "; }
    ~B() { cout << "b2 "; }
};

void test()
{
    B b;
}

int main()
{
    A a;
    test();
    return 0;
}
```

- 1) a1 b1 b2 a2
- 2) a1 b1 a2 b2
- 3) a1 a2 b1 b2
- 4) b1 b2 a1 a2
- 5) b1 a1 a2 b2
- 6) A compile error occurs

16. Write down the expected result of running the following program with the given standard input.

Student Number:

Name:

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

int main () {
    string str;
    getline(cin, str, ',');
    getline(cin, str, ',');
    getline(cin, str, ',');
    cout << str << endl;

    return 0;
}
```

The given standard input (↵ means pressing Enter key):

```
hello:world,good,bye,↵
```

17. What is the expected output of the following program?

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

int main()
{
    int a[] = {0,1,2,3};

    int* p = &a[0];
    (*p)++;
    p = &a[1];
    (*p)++;

    int& r = a[2];
    r++;
    r = a[3];
    r++;

    cout << a[0] + a[3] << endl;
}
```

- 1) 4
- 2) 5
- 3) 3
- 4) 6
- 5) A compile error occurs

Student Number:

Name:

18. What is the expected output of the following program?

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

int sum(int x, int y, int z=100, int w=10)
{
    return (x + y + z + w);
}

int main() {
    cout << sum(10, 20, 30) << endl;
    return 0;
}
```

- 1) 60
- 2) 140
- 3) 130
- 4) 70
- 5) A compile error occurs

19. What is the expected output of the following program?

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

namespace first_space {
    void func() {
        cout << "a";
    }
}

namespace second_space {
    void func() {
        cout << "b";
    }
}

int main () {
    using namespace first_space;
    func();

    using namespace second_space;
    func();

    return 0;
}
```

- 1) ab
- 2) a

Student Number:

Name:

3) b

4) A compile error occurs

20. Choose **ALL** that are **not appropriate** to fill in blank (a).

```
#include <stdio.h>
typedef struct
{
    int xpos;
    int ypos;
} Point;

int main()
{
    Point points[5];
    for(int i=0; i<5; ++i)
    {
        points[i].xpos = i;
        points[i].ypos = i*2;
    }

    for(int i=0; i<5; ++i)
    {
        printf("%d %d\n", points[i].xpos, _____(a)_____);
    }

    return 0;
}
```

- 1) points[i].ypos
- 2) (&points[i])->ypos
- 3) (&points[i]).ypos
- 4) (points+i)->ypos
- 5) (points+i).ypos
- 6) (*(points+i))->ypos
- 7) (*(points+i)).ypos

6 points

21. Select ALL the code that can be in the blank line _____ of the following code.

Student Number:

Name:

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

class Test{
public:
    Test(int a, int b) {}
    Test(int a) {}
    Test(char c) {}
    ~Test() {}
};

int main() {
    Test _____
}
```

- (a) test(1, 2);
- (b) test;
- (c) test(3);
- (d) test("a");
- (e) None of the above

22. Fill in the blanks so that the following program successfully compiles and runs without a memory leak.

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    cin >> n;

    _____ (a) _____ num = _____ (b) _____ int;
    _____ (c) _____ numArr = _____ (d) _____ int[n];

    *num = n;
    for(int i=0; i<n; i++)
        numArr[i] = i;

    cout << *num << endl;
    for(int i=0; i<n; i++)
        cout << numArr[i] << " ";
    cout << endl;

    _____ (e) _____ num;
    _____ (f) _____ numArr;

    return 0;
}
```


Student Number:

Name:

2)

```
print("creative");
```

b)

```
void print(double d, int width);
```

3)

```
print(2145L, 52);
```

c)

```
void print(long l, int width);
```

4)

```
print("design", 100);
```

d)

```
void print(int i, int width);
```

5)

```
print(220, 15);
```

e)

```
void print(const char* str);
```

6)

```
print("my name", 10.0);
```

f)

```
void print(const char* str, double d);
```

1): ____

2): ____

3): ____

4): ____

5): ____

6): ____

8 points

26. Fill in the blanks so that the following program successfully compiles and runs without a memory leak.

Student Number:

Name:

```

#include <iostream>

class Vector {
public:
    int* x;
    int size;

    Vector() : x(NULL), size(0) {}
    Vector(int n) { size = n; x = _____ (a) _____ int[size]; }

    ~Vector() { _____ (b) _____ x; }

    void Resize(int n) { _____ (c) _____ x; x = _____ (d) _____ int[size = n]; }
};

int main() {
    Vector* p = _____ (e) _____ Vector(10);
    for (int j = 0; j < p->size; ++j) p->x[j] = j;

    Vector* q = _____ (f) _____ Vector[10];
    for (int i = 0; i < 10; ++i) q[i].Resize(i);

    _____ (g) _____ p;

    _____ (h) _____ q;
    return 0;
}

```

27. You want to build a project consisting of two source files, “main.cpp” and “myfunc.cpp”, and write a CMakeLists.txt so that an executable file named “test” is created using the “make” command. Please fill in the blanks below in the CMakeLists.txt.

_____ (a) _____ (_____ (b) _____ _____ (c) _____ _____ (d) _____)

28. The following program takes a string (s) and an integer (n) as command line arguments and print out the string s n times, as shown in the example below. Modify the code to work correctly.

```

1: #include <iostream>
2: #include <string>
3: #include <stdlib.h>
4: using namespace std;
5: int main(int argc, char *argv[])
6: {
7:     for(int i=0; i<argv[1]; i++)
8:         cout << argv[0] << endl;
9:
10:     return 0;
11: }

```

Student Number:

Name:

Example of running the program:

```

$./test abc 3
abc
abc
abc

```

```

$./test abc 2
abc
abc

```

line 7:

10 points

29. Fill in the blanks so the following program will print the output below.

```

#include <iostream>
using namespace std;

class MyString {
private:
    string _str;
public:
    MyString():           (a)           { cout << "dctor" << endl;}
    MyString(const string& str) { _str = str; cout << "ctor" <<
endl; }
    string getString() { return _str; }
};

class Student {
public:
    MyString _name;
    int _id;
    Student():           (b)           {}
    Student(string name, int id):           (c)           {}
};

int main()
{
    Student st1;
    cout << st1._name.getString() << " " << st1._id << endl;
    Student st2("Tom", 2020);
    cout << st2._name.getString() << " " << st2._id << endl;
    Student st3("Bar", 2021);
    cout << st2._name.getString() << " " << st2._id << endl;
}

```

Output:

```

dctor
noname -1
ctor
Tom 2020
ctor
Bar 2021

```

Student Number:

Name:

30. The following program uses a STL list container to store data. Write down the print() function that fills in the blank (a) and (b) so that the following execution results are displayed when the program is executed.

```
#include <iostream>
#include <list>
using namespace std;
void printList(const list<int>& intL){
```

(a)

```
}

int main(){
    int n;
    cin >> n;

    list<int> intL;
    for(int i=0; i<n; ++i)
        (b)

    printList(intL);
    return 0;
}
```

(Example of running the program)

3↵

0 1 2

31. A palindrome is a word which reads the same backward as forward. For example, noon, level, kayak, etc. are palindromes. The isPalin() function below is a function that tests whether a given word is palindrome or not (returns true only if the input string s is palindrome). Fill in the blanks (a), (b) of the _isPalin() function for this to work. Also write down the output (c)-(f) when running.

Student Number:

Name:

```
bool _isPalin(const string& s, int begin, int end) {
    if (begin >= end-1) return true;
    return ( _____ (a) _____ ) &&
           ( _____ (b) _____ );
}

bool isPalin(const string s) {
    if (s.empty())
        return false;
    else
        return _isPalin(s, 0, s.size());
}

int main() {
    cout << "isPalin(\"abcdcdcba\")? " << isPalin("abcdcdcba") << endl;
    cout << "isPalin(\"ab\")? " << isPalin("ab") << endl;
    cout << "isPalin(\"a\")? " << isPalin("a") << endl;
    cout << "isPalin(\"\")? " << isPalin("") << endl;
}

// output:
//     isPalin("abcdcdcba")? _____ (c) _____
//     isPalin("ab")? _____ (d) _____
//     isPalin("a")? _____ (e) _____
//     isPalin("")? _____ (f) _____
```