

C/C++ Programming Sample Test 1

Question 1. (6 points) What is the purpose of testing a program with sample data as input?

To see if the program works as expected in all cases.

Question 2. (9 points) Indicate the resulting value by evaluating each of the following expressions.

a) $4.0 / 10.0 + 3.5 * 2$ Result: 7.4

b) $11 + 10 \% 3 + 2.1 * 3$ Result: 18.3

c) $!(4.0 < 10.0) || 3 >= 5$ Result: false

Question 3. (15 points) Complete a simple program that allows the user to enter the lengths of three sides of a triangle (a, b, c) and calculates the triangles area by the following formulas:

$$s = \frac{a+b+c}{2}$$

$$area = \sqrt{s(s-a)(s-b)(s-c)}$$

```
#include <cmath> // From the cmath module use the sqrt( ) function
```

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

```
double a, b, c, s, area;
```

```
int main() {
    cout << "Enter a:";
    cin >> a;
    cout << "Enter b:";
    cin >> b;
    cout << "Enter c:";
    cin >> c;
    s = (a + b + c) / 2;
    area = sqrt(s * (s - a) * (s - b) * (s - c));
    cout << "area = " << area << endl;
}
```

Question 4. (10 points) For the following program indicate the output for each of the inputs:

```

cout << "Enter three numbers: ";
cin >> a >> b >> c;

if (a > b) {
    cout << "Who?";
    if (b > c) {
        cout << "Dog";
    } else {
        cout << "Cat\n";
    }
} else if (b > c) {
    cout << "What?\n";
    if (a >= c) {
        cout << "One";
    } else if (a < b) {
        cout << "Two\n";
    } else if (c == b) {
        cout << "Three";
    }
} else {
    cout << "Where?";
    if (a == b) {
        cout << "Up";
    } else {
        cout << "Down\n";
    }
}
cout << "Done\n";
    
```

a)

Expected Output if a = 3, b = 4, and c = 5 are input
Where? Down\n Done

b)

Expected Output if a = 5, b = 4, and c = 3 are input
Who? Dog Done\n

c)

Expected Output if a = 3, b = 3, and c = 5 are input
Where? Up Done

Question 5. (10 points) Write C++ statement(s) to generate a random integer value between 4 and 18 (inclusive). Assume that the appropriate includes have been done, and the `rand()` function has been properly seeded.

~~std::rand()~~

random Number = `rand() % 15 + 4;`

Question 6. (15 points) Using any appropriate kind of if statements, write C++ code to output the appropriate string according to variable myTemperature's value.

myTemperature	String
myTemperature < 96	"Below normal"
96 <= myTemperature < 100	"Normal"
100 <= myTemperature <= 101.5	"Slight temperature"
101.5 < myTemperature	"High Fever"

```

if ( myTemperature < 96 ) {
    cout << "Below normal \n";
} else if ( myTemperature < 100 ) {
    cout << "Normal \n";
} else if ( myTemperature <= 101.5 ) {
    cout << "Slight temperature \n";
} else {
    cout << "High Fever \n";
} //end if
    
```

Question 7. (10 points) What would be the expected output of the following program?

```

for ( i = 1; i <= 5; i++ ) {
    for ( j = 1; j <= 4; j++ ) {
        cout << "j ";
    } // end for

    cout << endl;

    for ( k = i; k <= 6; k++ ) {
        cout << "k ";
    } // end for
} // end for
    
```

```

j k j k j k j k \n
k k k k j j j j \n
k k k k j j j j \n
k k k j j j j \n
k k k
    
```

Question 8. (10 points) Consider the following menu code.

```
const int SIZE = 50;
char grade[SIZE];
. . .

cout << "Math Tutor Menu" << endl << endl;

cout << "1.  First Grade" << endl;
cout << "2.  Second Grade" << endl;
cout << "3.  Third Grade" << endl << endl;

cout << "Enter your grade (1, 2, or 3): ";
cin.getline(grade, SIZE);
```

Why would reading the grade as a string be better than reading it as an integer (int variable)?

If the user entered "first" instead of 1, then we could not read it into an int variable grade.

Question 9. (15 points) Correct the logic errors in the following code that should calculate the average test score. The while loop is sentinel-controlled with any negative test score signaling the end of the input. The sentinel value should not be included in the average calculation.

```
int counter;
double average, total, nextScore;
```

```
counter = 1 0;
total = 0 0;
```

```
cout << "Enter a test score (or -1 when done): ";
cin >> nextScore;
```

```
while (nextScore >= 0 >= 0) {
    total + nextScore;
    counter + counter + 1;
    cout << "Enter a test score (or -1 when done): ";
    cin >> nextScore;
```

total = total + nextScore;

```
} //end while
if (counter == 0) {
total = total + (nextScore)
} else {
    average = counter / total;
    cout << "The test average was " << average << endl;
}
```

total / counter;