CS 101, Fall 2022 - Quiz 1
Sat, 26 Nov 2022, 12:30 to 14:30

Instructions

- Please read these instructions carefully before you start your exam.
- Keep your ID card on the table for ready reference
- Make sure that you sign the attendance sheet
- The quiz contributes to 11.25% of your final total for this course.
- There are 25 questions in this quiz. All questions are compulsory and carry one point each for a total of 25. Read each question carefully before you answer. For multiple choice questions, you must enter **one and only one answer** per question.
- There is NO partial marking for any question. There is also NO negative marking.
- Answer all questions on your own. Do not interact with anyone else during the exam for the solutions.
- Please note that this is a closed book written test and hence, notes, cheatsheets, anything written on a paper, mobile phones, laptops, tablets, etc., are NOT allowed. All of these need to be kept in a bag which should be placed near the instructor's platform. Kindly note that calculators are also NOT allowed.
- Use the reverse side of the Answer sheets or question paper for your rough work
- Two answer sheets will be supplied to all of you.
- Upon receiving these, fill up your roll number, name, group number, and tick which day you attend the lab (Wed/Fri) on **both** the answer sheets.
- When you write your answers later, **you must write the same answer on BOTH the sheets, for each question that you attempt.**
- At the end of the test, the invigilators will collect **one** of your answer sheets (marked as **<u>Instructor's Copy</u>**) while you have to retain the **other** (marked **Student's Copy**).
- During the last 15 minutes please remain seated, even if you have finished answering.
- Please do not leave the classroom till all papers are collected, and the count is tallied.
- Please turn over this sheet to view the questions

**~~~ All the best ~~~**

Roll Number: ______________ Name: ________________ Group#: ______

1. Consider the following program:

```
int main () {
     float score;
     cout << "enter your score: "; cin >> score;
     if (score > 90.0) cout << "You got an AA";
     else if (score  > 85 && score <= 90)
          cout << "You got an AB";
     else if (score > 80 && score <= 85)
          cout << "You got a BB";
     else cout << "You scored less than a BB";
}
```

   Consider the statements:
   (S1) You can exclude the condition `score <= 90` in the `else if` statement for `"You got an AB"`, without changing the output for any value of score entered by the user
   (S2) You can exclude the condition `score <= 85` in the `else if` statement for `"You got a BB"`, without changing the output for any value of score entered by the user

   Which of these statements is/are true?
   (A) Only S1
   (B) Only S2
   (C) Neither S1 nor S2
   (D) Both S1 and S2

2. The output of the following program is _____.

```
int main(){
     int a = 10;
     for(int i = 0; i< 2; i++){
        int a = 4;
        a = a + 1;
        cout << a;
     }
     cout << a;
}
```

3. Consider the number 12 expressed in decimal representation. Its binary representation is _____.

4. Consider the following statements regarding the difference between `while` and `do while` loops. Which of the following statements is true?

(A) `do while` loops cannot contain `break` or `continue` statements, whereas `while` loops can

(B) The body of a `do while` loop is always executed at least once whereas the body of a `while` loop may never be executed even once

(C) A `for` loop cannot be nested inside a `do while` loop whereas it can be nested inside a `while` loop.

(D) Variables cannot be initialized inside the body of a `do while` loop whereas they can be initialized inside the body of a `while` loop.

5. What is the output of the following code fragment?

```
float f = (1/9)*9;
if (f==0) cout <<f<<"The computer never makes a mistake";
else cout <<f<<"The computer can make a mistake";
```

(A) 1 The computer can make a mistake

(B) 0 The computer never makes a mistake

(C) 0.9999 The computer can make a mistake

(D) 0.9 The computer can make a mistake

(E) 0 The computer can make a mistake

6. Out of the following, which datatype is ideal to represent the number of cafes in Mumbai, in a computer?

(A) float

(B) unsigned int

(C) signed int

(D) bool

(E) char

7. Which statement in the following code fragment causes a compiler error? The statement numbers (S1,S2,S3,...,etc) are labeled as comments in the code fragment

```
   int u; char v; float w; const float x = 3.14; unsigned int y; //
S1
   u = ‘a’; // S2
   v = 3; // S3
   w = 3; // S4
```

```
u = 3.14; // S5
x = 3.14; // S6
y = -1; // S7
```

(A) S7 only
(B) S6 only
(C) Both S7 and S6
(D) S3 and S2
(E) None of these

8. The output of the following code fragment is __________.

```
int a,m,n;
m = 10; n = 11;
a = ++m-n--;
cout << m << n << a;
```

9. The output of the following code fragment is __________.

```
int x,y,z;
x = y=z=6;
x += y -= z *= 2;
cout << y << z << x;
```

10. Consider the following statements regarding the use of a watchpoint in GNU debugger (gdb):

(S1) It is used to temporarily stop execution when the value of a chosen variable changes
(S2) It is used to temporarily stop execution when there is an attempt to change the value of a chosen set of memory locations
(S3) It is used to temporarily stop execution when the value of a chosen expression changes

Which of the above statements is/are true?
(A) S1, S2 and S3
(B) All three are false
(C) S1 only
(D) S2 only
(E) S3 only

11. The code fragment can be replaced by which of the ones that follow? You can assume that m,a,b are all integers.

```
m = a < b ? a : b;
```

(A) `if (a < b) m = b; else m = a;`
(B) `if (a < b) m = a; else m = b;`
(C) `if ( a ==b) m = a;`
(D) `m = a;`
(E) None of these

12. What is the output of the following code fragment?

```
int A=0, B=10;
if (A == 0)
   if (B == 0) cout << "A and B are both zero";
else cout << "A is not zero";
cout<<"*Seeyou";
cout <<"*Goodbye";
```

(A) A is not zero*Seeyou*Goodbye
(B) A and B are both zero*Seeyou*Goodbye
(C) No output is printed
(D) *Seeyou*Goodbye
(E) *Goodbye

13. Consider the following code snippet which tries to test whether the point (tx,ty) (as generated in the code snippet) lies on the line passing through (x1,y1) to (x2,y2)

```
main_program {
    double x1,y1,x2,y2;
    double m,c;
    double tx,ty,errv;

   // starting point (x1,y1) of the line segment
    x1 = 3.0;
    y1 = 9.0;

    // ending point (x2,y2) of the line segment
    x2 = sqrt(2.5)*x1+sqrt(3.5);
    y2 = sqrt(2.5)*y1+sqrt(3.5);

   // computing slope m and intercept c of the line
```

```
    m = (y2-y1)/(x2-x1);
    c = y2-m*x2;

    // generating a point (tx,ty) on the line
    tx = x1+0.1*(x2-x1);
    ty = y1+0.1*(y2-y1);

    // error value: does (tx,ty) lie on the line?
    errv = ty-m*tx-c;

    if (errv == 0)
       cout << "The point (tx,ty) lies on the line";
    else
       cout << "The point (tx,ty) does not lie on the line";
}
```

What is the main problem with the program regarding the manner in which it tests whether or not (tx,ty) lies on the line passing through (x1,y1) and (x2,y2)?

(A) The program does not test whether (tx,ty) lies in between (x1,y1) and (x2,y2) even though it **accurately** tests whether (tx,ty) lies on the line passing through those two points
(B) The program unnecessarily uses double instead of float
(C) The test whether (tx,ty) lies on the line passing through (x1,y1) to (x2,y2) is based on comparing a floating point number to zero, which may fail due to numerical precision issues
(D) The method of slope computation in the program is incorrect
(E) The method of intercept computation in the program is incorrect

14. Consider the code fragment

```
if (!(c==0 && d == 20)) cout << “Hello”;
```

The condition in the `if` statement can be changed to which of the following, without changing the output for any value of `c` or `d` ?

(A) `c !=0 || d!=20`
(B) `c !=0 && d!=20`
(C) `c !=0`
(D) `d!=20`
(E) `None of these`

15. What is the output of the following code fragment?

```
int a = 1, b = 10;
while (a = b)  { a = a+1; b = b -1;}
cout << a <<”,” << b;
```

(A) 1,10
(B) The program goes into an infinite loop
(C) 10,1
(D) 10,0
(E) 0,0

16. What is the output of the following code fragment?

```
int a = 1, b = 10, c = 0;
while (a = b)  {c++;}
cout << a <<”,” << b << “,” << c;
```

(A) `1,10,0`
(B) `10,10,32768`
(C) `The program enters an infinite loop`
(D) `1,10,32768`
(E) `10,10,0`

17. How many times will the following two code fragments print `Welcome`? Note that the range of a short integer (`short int`) is from -32,768 to +32,767 whereas that of an integer (`int`) is -2,147,483,648 to +2,147,483,647.

Fragment 1:

```
int i;
for (i=0;i<32800;i++) cout << “Welcome”;
```

Fragment 2:

```
short int j;
for (j=0;j<32800;j++) cout << “Welcome”;
```

(A) Fragment 1 will print Welcome 32799 times; Fragment 2 will print Welcome 32799 times
(B) Fragment 1 will print Welcome 32800 times; Fragment 2 will print Welcome 32800 times
(C) Fragment 1 will print Welcome 32800 times; Fragment 2 will produce an infinite loop
(D) Fragment 1 will print Welcome 32799 times; Fragment 2 will produce an infinite loop
(E) Fragment 1 will print Welcome 32801 times; Fragment 2 will print Welcome 32801 times

18. Consider the following code snippet:

```
a = 0; x = 10; y = 20; z = 30;
for(int i=0; i<n; ++i) {
    x = y+z; a = a + 1000*6*i + x*x;
 }
```

Consider the following expressions:

```
(S1) x=y+z;
(S2) 1000*6
(S3) x*x
```

Which of the expressions can be moved out of the `for` loop (possibly also including extra variables to store their value) to produce an equivalent code which would execute faster? For example, if you decided to move `x*x` out of the `for` loop, you could write a statement `int q = x*x;` before the `for` loop and use just the value of `q` inside the `for` loop instead of `x*x`.

(A) `S1 only`
(B) `S2 only`
(C) `S3 only`
(D) `All three: S1, S2, S3`
(E) `Neither S1 nor S2 nor S3`

19. (multi-part question) Consider a sequence of numbers of the form f(n) = f(n-2) + f(n-3), i.e. the n-th number is equal to the sum of the (n-2)-th and (n-3)-th numbers, where the first three numbers in the sequence, i.e. f(0), f(1), f(2) are all equal to 1. The code snippet below is intended to print all the numbers of this sequence from f(0) till f(15) (both inclusive). However, the code snippet is incomplete. You expected to fill in the blanks as directed:

```
int pp = 1, p = 1, curr = 1, next = 1;
cout << pp << " " << p << " " << curr << " ";
for(int i = 0; i <= ______(i)______; i++) {
    next = ______(ii)______;
    pp = p;
    p = ______(iii)______;
    curr = ______(iv)______;
    cout << next << " ";
}
```

In place of (i), the correct number is _______.

20. (multi-part question) Referring to the same program as earlier, the correct statements for (ii), (iii), (iv) respectively are:

(A) `next = pp+p; p=curr; curr=next+1;`

(B) `next = pp+p; p = next; curr = next;`

(C) `next = pp; p = curr; curr = next;`

(D) `next = pp; p = next; curr = p;`

(E) `next = pp+p; p = curr; curr = next;`

21. Consider the following code fragment involving a `for` loop.

```
for(i=0;i<5;i++) if (i == 2) continue;
```

An inquisitive student (and hence a very good student! :-)) wishes to rewrite the code using a `while` loop. What statement(s) XYZ inside the `while` loop below will make his/her code fragment equivalent to that using the for loop?

```
i = 0;
while (i < 5) { XYZ }
```

(A) `if (i == 2) continue; i++;`

(B) `if (i == 2) i++;`

(C) `i++;`

(D) `if (i!=2) i++;`

(E) `if (i == 2) {continue; i++;}`

22. A student has written the following function:

```
int myfunction (unsigned short int k) {
   int i= 0;
   while (i*i*i <= k) i = i+1;
   return (i-1);
}
```

Which of the following best describes `myfunction` if input 0<=k <= 65535? Note that the range of `unsigned short int` is from 0 to 65535.

(A) It returns the floor of the cube root of k

(B) It returns the ceiling of the cube root of k

(C) It returns the exact cube root of k

(D) It will fall into an infinite loop and hence it is difficult to describe what it exactly does

(E) It returns the square root of k

23. A student has written the following function:

```
int myfunction (int k) {
   int i= 0;
   while (i*i*i <= k) i = i+1;
   return (i-1);
}
```

The output of this function with a *negative* input parameter k is:

(A) the floor of the cube root of k

(B) The ceiling of the cube root of k

(C) -1

(D) 0

(E) It will fall into an infinite loop and hence it is difficult to describe what it exactly does

24. What is the problem with the following function which seeks to find the factorial of a positive integer? Assume that the argument `n` given to the function is always non-negative.

```
int factorial (int n) {
     int f;
     if ( n == 0 || n == 1) return 1;
     while (n > 1) {f = f*n-- ; }
     return f;
}
```

(A) The program will produce a syntax error in the statement `f = f*n-- ;`

(B) The program will loop infinitely

(C) The output of the program will be unpredictable because `f` was not initialized to 1 before starting the `while` loop for computing the products

(D) Both (A) and (C)

(E) There is no problem with this program

25. Consider a function below:

```
int myfn1 (int n) {
   return 2*n;
}
```

The statement `m = myfn1(m);` has the effect of doubling the value of parameter `m`. Consider the following ways to rewrite this function as a void function `myfun2` with parameter `m` such that the same effect of doubling the parameter value is achieved by the statement `myfun2(m)`.

```
1. void myfn2 (int &n) {
       n=2*n;
   }

2. void myfn2 (int n) {
       n=2*n;
   }
```

Out of these which is/are the correct ways to rewrite the function? Note that the function call will be of the form `myfun2(m);`

(A) 1 only
(B) 2 only
(C) Neither out of 1,2
(D) Both 1 and 2

Answers:

1. (D) Both S1 and S2 are true
2. Answer is 5510
3. 1100
4. B
5. The computation in `float f = (1/9)*9` assigns the value of 0 to f as (1/9) is treated as 0. Hence the correct answer is (B) "0 The computer never makes a mistake". A different output would be produced with (1.0/9)*9 or (1/9.0)*9 or (1.0/9.0)*9.
6. B
7. (B) S6 is the correct answer as you cannot assign a value to a variable declared as a constant (even if it is the same value as it was initially assigned to). Assigning a character value to an integer will just assign the ASCII value to the integer. Assigning an integer value to a float or a negative integer value to an unsigned integer also will not cause any syntax errors.
8. 11100 is the correct answer. This is because m is first incremented to 11. Then n = 11 is deducted from it to produce a = 0. Lastly n is decremented to produce 10.
9. -6120 is the correct answer. z=6 is first multiplied by 2 to produce z=12. From y = 6, z is deducted to produce y = -6. Lastly y=-6 is added to x=6 to yield x = 0.
10. A
11. (B) The right answer is `if (a < b) m = a; else m = b;`
12. (A) A is not zero*Seeyou*Goodbye is the correct answer as the else is associated with the second if loop.
13. The correct answer is (C), i.e. the test is wrong due to numerical precision issues
14. By demorgan's law (A) `c !=0 || d!=20` is the right answer.
15. (E) - (0,0) is the correct output.
16. The correct answer is (C) infinite loop
17. Option (C) Fragment 1 will print Welcome 32800 times; Fragment 2 will produce an infinite loop is the correct one. The first for loop indeed runs until i = 32799, which means it prints welcome 32800 times. In case of the second loop, short int has values ranging from -32768 to 32767. When short j = 32767, then j++ will yield -32768 due to which this becomes an infinite loop.
18. The correct answer is (D) All three S1, S2, S3
19. The correct answer is 12 as the first three numbers of the sequence were already printed earlier
20. The correct answer is next = pp+p; p = curr; curr = next; (option E)
21. (C) i++ is the right answer
22. (A) This returns the floor of the cube root of k
23. (C) -1 is the correct answer as it will simply not enter the while loop due to which i remains 0 and i-1=-1 is the output.
24. C
25. A. Only method 1 is right