

CS 101 Autumn 2023 — Theory Quiz 1

30 August 2023

8 questions, 70 marks

(Instructor: Suyash P. Awate)

Roll Number	
Division and Group	
Name	

Q. No.	Marks	Graded By	Verified By	Student Remarks / Crib
1				
2				
3				
4				
5				
6				
7				
8				
TOTAL				

Please read the following instructions carefully before you start.

- Write your roll number, name, and group number on this page in the space provided. A paper without a roll number and name will NOT be graded.
- Write your answers neatly with a blue/black pen on this question paper itself in the space provided for each question. At the end, you must submit this paper to the invigilator.
- Rough pages will NOT be provided. Use the empty space in the margins.
- Please note that your answers should NOT include any programming concept that hasn't been covered in the class so far. If such answers are found, they shall NOT be graded.
- No clarifications will be provided on any question. When in doubt, make suitable assumptions, state them clearly, and proceed to solve the problem.
- All the best! 🍊

1. [10 points]

- (3 points) Convert 1.11 from binary to decimal.

1.75 or $\frac{7}{4}$

- (7 points) Convert 1.11 from decimal to binary, accurate upto 8 places after the radix point.

round $(1.11 * 2^8) = \text{round}(284.16) = 284$
284 in binary = 100011100
so answer = 1.00011100

2. [12 points]

For each of the following four expressions, assume that m is an integer variable with value 25 and n is an integer variable with value 7. Then, do the following:

(i) evaluate the expression and state the resulting value of the expression;

(ii) if, after evaluating the expression, the value of m or n changes from its initial value (i.e., 25 or 7), then indicate the new value(s) of the variable(s); else indicate that the value is unchanged. **1 mark for each correct block/table cell**

No.	Expression	Value of the Expression	value of m after the expression is executed	value of n after the expression is executed
1	$m = n = 3$	3	3	3
2	$m \% n++$	4	25 or unchanged	8
3	$m \% ++n$	1	25 or unchanged	8
4	$++m - n--$	19	26	6

3. [10 points]

You want to write a program that does all of the following:

Task A (already done in the code below): ask the user to input the end points of a line segment in a 2-dimensional Euclidean plane;

Task B: read the coordinates into variables named x_1, y_1, x_2, y_2 ;

Task C: compute the length of the line segment and store it in a variable named *length* and output the value on screen;

Task D: compute the slope of the line segment and store it in a variable named *slope* and output the value on screen.

In the code below, fill in the missing statement at each location indicated by "...".

```
main_program

// Task A
cout
<< "enter coordinates of end points of line segment in order x1, y1, x2, y2"
<< endl;

// Task B
// 1 statement (not more) to define variables (2 points)
float x1, y1, x2, y2;

// 1 statement (not more) to read in variables (2 points)
cin >> x1 >> y1 >> x2 >> y2;

// Task C
// 1 statement (not more) to define variable (1 point)
float length;

// 1 statement (not more) to compute length (2 points)
length = sqrt ( pow (x1 - x2, 2) + pow (y1 - y2, 2) );

cout << "length: " << length << endl;

// Task D
// 1 statement (not more) to define variable (1 point)
float slope;

// 1 statement (not more) to compute slope (2 points)
slope = (y2 - y1) / (x2 - x1);

cout << "slope: " << slope << endl;
```

4. [10 points] (4 + 6)

Describe precisely the output of the following program after each iteration of the repeat loop. There are two kinds of output, (i) text and (ii) drawing. For the drawn output, draw the output yourself after each iteration, specifying the lengths and angles exactly.

```
#include <simplecpp>
main_program
{
    turtleSim();
    unsigned int i = 0;
    repeat (4)
    {
        cout << "i: " << i << endl;
        left (90);
        forward (+ 50 * cosine (i* 10));
        forward (- 50 * cosine (i* 10));
        right (90);
        forward (50);
        repeat (18) i++;
        wait (2);
    }
}
```

Value of variable i that is printed using cout (1 mark each)

- 1st Iteration: 0
- 2nd Iteration: 18
- 3rd Iteration: 36
- 4th Iteration: 54

Initial position of the turtle is:



Drawing

1 mark each for each correct drawing. 0.5 marks each for each correct length. No deduction for not writing the angle.

Drawing after 1st Iteration	Drawing after 2nd Iteration
Drawing after 3rd Iteration	Drawing after 4th Iteration

5. [12 points]

Go through the program given below and fill up the blanks. (1) The program should first accept the total number of digits in a number. (2) Then accept the number, digit by digit. (3) Finally, compute the reverse and print it. An example input/output is given below:

```
Enter number of digits: 4
Enter the digits one by one
1 2 3 4
The reverse is: 4321
```

```
#include<simplecpp>
main_program {
    int digit, numdigits;

    int ans = ____blank A____;
    int temp = ____blank B____;

    cout << "Enter number of digits: ";
    cin >> numdigits;
    cout << "Enter the digits
            one by one" << endl;

    repeat(____blank C____) {
        cin >> digit;
        ans = ____blank D____;
        temp *= 10;
    }
    cout << "The reverse is: " << ans;
}
```

3 marks for each correct part

blank A: 0

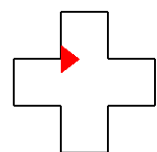
blank B: 1

blank C: numdigits

blank D: ans + digit*temp

6. [6 points] **3 marks for each correct option.**

Go through the program given below and choose **ALL correct options** such that the program draws a plus sign as shown on the right. The initial and the final position and orientation of the turtle is as shown in the picture. **0 marks if any wrong option is chosen along with correct one**



```
#include<simplecpp>
main_program {
    turtleSim();
    repeat(4) {
        ____blank A____;
        forward(50);
        repeat(2) {
            ____blank B____;
            forward(50);
        }
        wait(2);
    }
}
```

(a) blank A: left(90) and
blank B: right(90)

(b) blank A: left(270) and
blank B: right(-270)

(c) blank A: left(-270) and
blank B: right(270)

(d) blank A: right(270) and
blank B: right(-270)

Write your Answer:

**a: left(90), right(90) and
d: right(270), right(-270)**

7. [4 points]

Go through the program given below and identify at least four errors. Describe the error against that appropriate line in the adjacent space provided. If a particular line does not have an error, leave it blank. **Each correct error (with proper reasoning) gets 1 mark, ignore all other errors (no negative marks)**

Program	Describe the errors
#include<simplecpp>	
main program {	main program should be main_program
int a = 5, b = 6, c = 7;	
float b = 5.345;	conflicting declaration as b is already declared
INT fact = 120;	INT should be int
cout << "Enter a number" << endl;	
cin << c;	Incorrect redirection operator. cin << c should be cin >> c
char c = 'H';	variable c is already declared
a + b = c;	lvalue required as left operand of assignment
}	

8. [6 points]

Go through the program given below.

```
#include<simplecpp>
main_program {
    int n1, n2, n3, n4;
    cin >> n1 >> n2 >> n3;
    n4 = n2;
    cout << n1 << " ";
    repeat(n3-1) {
        cout << n1 * n4 << " ";
        n4 = n4 * n2;
    }
}
```

What is the output of the following program if: **Two marks for each section, provided all the answers in a section are correct**

n1 = 4, n2 = 4, n3 = 4
4 16 64 256

n1 = 2, n2 = 3, n3 = 7
2 6 18 54 162 486 1458

n1 = 7, n2 = 6, n3 = 5
7 42 252 1512 9072



End of the paper

