**Student Name:** **Weight:**

**Student ID:** **Marks: 2% - BONUS**

**CREATE AN EMPTY DOCUMENT TO SUBMIT YOUR SOLUTIONS. DO NOT USE THIS DOCUMENT TO SUBMIT YOUR ANSWERS. YOU WILL LOSE 10% FOR DOING SO!!!**

Assignment FOR BONUS MARKS:

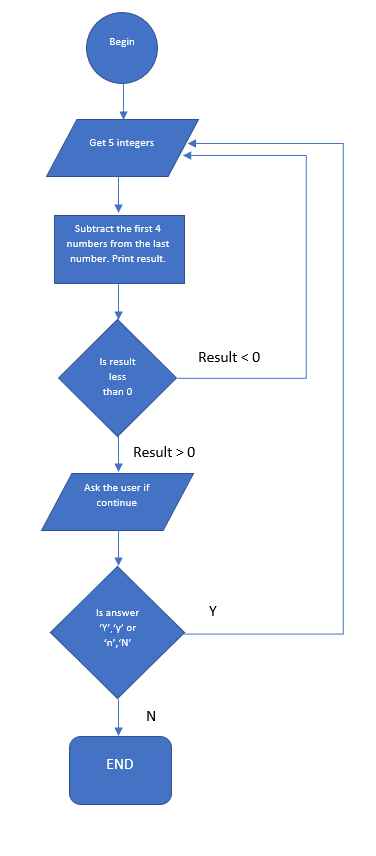
**Important:**

* You can certainly reach out for help.
* My one request is that you must have done some work, thought about the problems prior to reaching out.
* You are not allowed to say:
  + **“I don’t understand what you mean by \_\_\_\_\_\_\_\_\_\_\_\_**”.
* Now I expect that you will say:
  + **“Did you mean I should \_\_\_\_\_\_\_\_\_\_\_\_\_\_ because the question said \_\_\_\_\_\_\_\_\_\_\_\_**.
* Here’s an example:

**Instruction**: You should get 100 numbers from the user.  
**Question**: I don’t understand what you mean by get 100 numbers, how should I get 100 numbers?  
**Better Questions**: You said get 100 numbers, I will use scanf; but should I get integers, floats? Can I use an array to store those values?

Problem 1:

1. Review the following flow chart. Based on the chart, generate the C code that matches the flow chart.
2. Currently the program does **NOT** output a message when it is exiting. **How would you change the flow chart to show that a message is printed before exiting the program**?



Problem 2:

1. Fix the following snippets so that the loops will run:

int arr(10) = (0);

for(int count; count < 100, count+1)

{

    print('arr(%d) = %d', arr(count));

}

float arr[10] = {0};

for(int count; count < 100; count+1)

{

    printf("'arr(%d) = %d'",count, arr[count]);

}

int arr[10] = {0};  // store for the random numbers

for(int count=0; count < 100; count+=1)

{

    int count = (rand() % 100) + 1; // Get 100 random numbers

    arr[count] = count; // store the random numbers into the array

    count += 0; // sum up all the random numbers

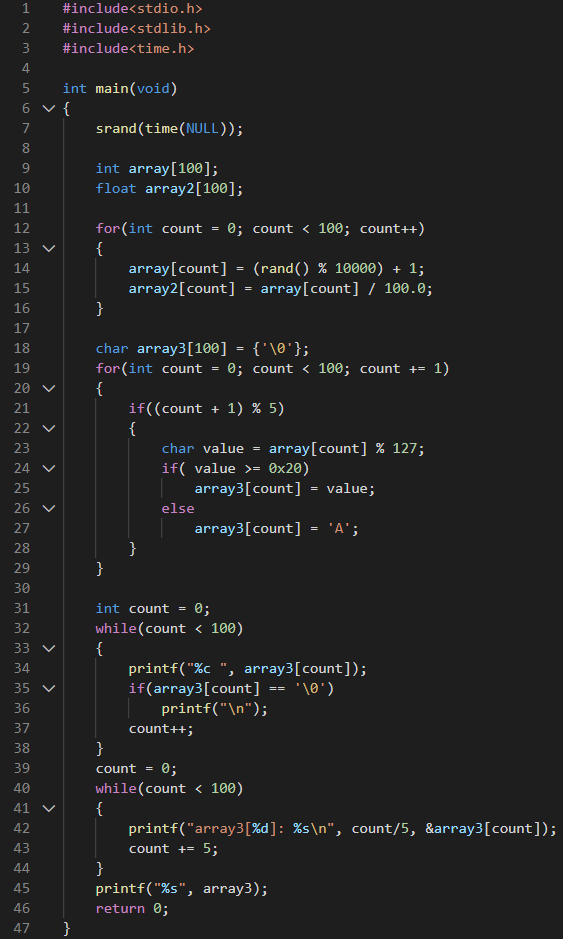
    int average = count / 100;  // Calculate the average of 100 numbers

    printf("average = %.2f", average);

}

Problem 3:

Analyze the following code and submit a list explaining your understanding of the code:



**Approach to analysis:**

1. What does the loop from lines 12-16 do?

**Write down a description of what you think is happening.**

1. What does the loop from lines 18-29 do?

**Write down a description of what you think is happening.**

1. What does the loop from lines 32-38 do?

**Write down a description of what you think is happening.**

1. What does the loop from lines 40-44 do?

**Write down a description of what you think is happening.**

1. Explain the output from all the printf statements.

**Saying that it prints a string and a number is NOT sufficient. Get deeper by showing your understanding of why that particular output is presented.**

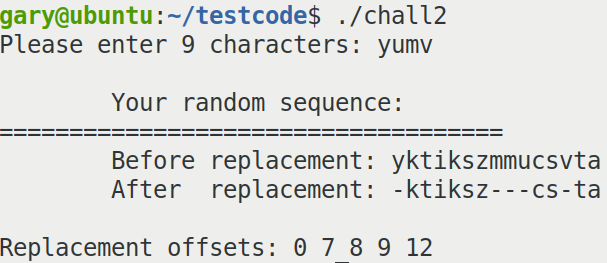
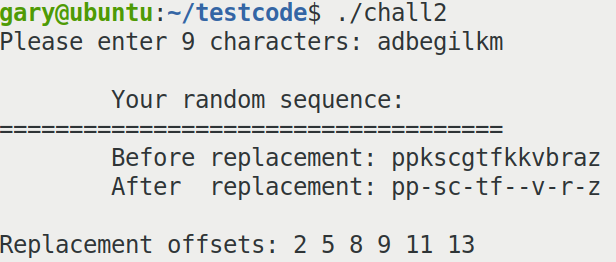
1. **Example Analysis:**
   1. *Line 11 is storing 100 random numbers between 1 and 10000 into the integer array named* ***array****.*

You can now proceed with the rest of the analysis starting with Lines 12.

Problem 4: (can be presented after class via recording)

Write a C program that does the following:

1. Has **getString** function to take up to **Four,** **9** character sequences from the user, *think multi- dimensional array arr[****4****][****10****]*, and places them into an array.
2. Has **fillArray** function to provide another array with **15** random characters **(a-z or 0-9)**.
3. Has **checkArray** function to determine if any of the random characters is in the strings provided by the user.
   1. You will replace the matching random characters with **‘-’** in your user entered string.
   2. You will also create an integer array that will save the offset where the characters were replaced. The below examples below show what the program sequence would be for a single dimensional array.



1. Has **printArray** function to print each row of the 4x10 array.