CSCI3100 Software Engineering

Assignment 6

Due – 11:59:59pm, 2nd May 2021 (Sunday)

Please submit the homework online through Blackboard.
Late submission penalty within 24 hours: 50%; after 24 hours: 100%.
Remember to go through Veriguide for Academic Honesty Declaration.
Missing Veriguide report: 50% mark deduction.

Answer the following problems based on lecture Topic 6 notes. Each question is assigned 25 points.

1. **Program Testing (25 points)**

The following C/C++ program is required for program flow analysis in this assignment:

```
1.  #include<string>
2.  #include<math.h>
3.  int test(int a, int b) {
4.      int a_array[32] = {0};
5.      int b_array[32] = {0};
6.      int c_array[32] = {0};
7.      int c = 0;
8.      if (a < 0) {
9.          a_array[0] = 1;
10.         a = -a;
11.     }
12.     if (b < 0) {
13.         b_array[0] = 1;
14.         b = -b;
15.     }
16.     int i = 31;
17.     while (a > 0 || b > 0) {
18.         a_array[i] = a%2;
19.         b_array[i] = b%2;
20.         a = a / 2;
21.         b = b / 2;
22.         i--;
23.     }
24.     for (int j = 31; j >=0; j--) {
25.         c_array[j] = c_array[j] + b_array[j] + a_array[j];
26.         if (c_array[j] > 1) {
27.             c_array[j] = c_array[j] - 2;
28.             c_array[j - 1] = 1;
```

```
29.             }
30.         }
31.         for (int j = 1; j < 32; j++)
32.             c += c_array[j] * pow(2, 32 - j - 1);
33.         if (c_array[0] == 1)
34.             return -c;
35.         else
36.             return c;
37.     }
```

(1) Draw the control flow graph of this program. Clearly identify each node with a number in the graph. (You can lump sequential statements into one node. The procedure partition is treated as one node without further analysis.)

(2) Calculate the following metrics of the above program according to the control flow graph (CFG) that you draw.
   a) # nodes
   b) # edges
   c) # predicate nodes
   d) # regions
   e) V(G) (McCabe's cyclomatic complexity number)

(3) Please give the equation to describe the relationship among the above 5 metrics.

(4) Please list all the linearly independent paths in the control flow graph of this program.

2. White-box Testing (25 points)

For the program shown in **Problem 1**:

(1) Can we design a test set for 100 % statement coverage (C_0)? If yes, please give the test set, using examples. If no, please give the reason. Important: The test set should be minimal.

(2) Is there any test set whose size is the same as the one in (1) can achieve 100% edge coverage (C_1)? If yes, please give such kind of test set. If no, please give the reason and provide additional test case(s).

(3) Is there any test set whose size is the same as the one in (2) can achieve 100% condition coverage (C_2)? If yes, please give such kind of test set. If no, please give the reason and provide additional test case(s). Is the answer to this question the same as or different from the previous one in (2), i.e., edge coverage (C_1)? Explain why.

(4) Can we achieve 100% path coverage (C_3) for this program? If yes, please describe the test set, and count the minimal number of test cases in the test set. If no, please provide the main reason why 100% path coverage cannot be achieved for this program.

3. Black-box Testing (25 points)

For the program shown in **Problem 1**:

(1) The programmer who develops this program tried to achieve the simulation of a value operation for one specific data type at the computer hardware level. Unfortunately, this program contains at least four bugs due to the programmer's poor knowledge of computer science. One of the bugs will lead to wrong output for more than half of the inputs. Two of the bugs will make the program fail in some test cases. The last bug will not affect the program in most cases but there is still a very rare chance to make the program throw an exception. Please explain what these bugs are. Note you don't need to fix them in the program.

(2) If previous bugs are fixed, could you please explain what kind of function does this program try to achieve?

(3) Please design a comprehensive test set to perform black-box testing. State your testing strategy for each of the selected test case in the test set with a clear description, and give an example of the corresponding test case. You need to list at least five test cases with different types. (Using a table would be recommended for a clear explanation)

(4) Can black-box testing and/or white-box testing find the bugs in (1)? Please give your judgement.

4. Stub and Driver Modules (25 points)

Nowadays live streaming is a very popular topic among young people. Your friends decide to develop a **live streaming platform** as their first business and invite you to join their team. You are assigned to develop *the income calculation module*. The whole system is described below:

i. A webcaster needs to open a website, which is generated by *the GUI module*, to input a user account and password.

ii. *The GUI module* will send the information to *the account management module* to check if the input information is correct.

iii. Once the input information is verified, *the account management module* will allow the webcaster to operate the user account.

iv. After successful login with the user account, the webcaster turns on *the live streaming module* to start live streaming. Every live streaming will last for two hours.

v. *The live streaming module* will count the number of customers and the gift value sent by the customers after the webcaster finish the user's live streaming. This recorded information will be sent to *the data recording module* by *the live streaming module*.

vi. The webcaster can choose to cash out from any live streaming the user has made. Once the webcaster made the decision, the account information will be sent to *the income calculation module* by *the account management module* and the data of the corresponding live streaming will also be sent to *the income calculation modul*e by *the data recording module*.

vii. Once *the income calculation module* gets the information from *the account management module* and *data recording module, the income calculation module* will calculate the income and send the information to *the transaction module* and *emailing module*.

viii. *The transaction module* will transfer the required money to the webcaster's bank account when *the transaction module* receives the transaction requirement from *the income calculation module*. *The emailing module* will send a confirmation letter to the webcaster's mailbox when this module receives the information from *the income calculation module*.

(1) You have finished your work of *the income calculation module*, but all other modules are not finished yet. Now you need to test your module. Please classify all other modules as either stub module, driver module, or not related for your testing, and design the corresponding stub modules and driver modules to help you finish the testing work.

(2) The income of a webcaster from single live streaming can be divided into two parts: customer rewards and gift rewards. The webcaster can get 2 HK dollars for every 100 customers. The rest customers will not be counted if the number is less than 100. To encourage the webcasters who don't have many fans, the platform set different gift rewards ratios for the different number of customers. The webcaster can get 50% of the gift value if his customers are less than 10,000 (0~9999). The webcaster can get 40% of the gift value if his customers are between 10,000 and 100,000 (10,000~99,999). The webcaster can get 35% of the gift value if his customers are between 100,000 and 1,000,000 (100,000~999,999) and the webcaster can get 30% of the gift value if his customers are

more than 1,000,000 (1,000,000~). The following table gives the simulated input for the number of customers and gift value for your income calculation module. Please give the expected outputs of these simulated inputs.

Number of customers	Gift value	Expected income
8,256	520	
12,863	2,753	
125,742	8,641	
1,246,732	12,845	

(3) According to the big data analysis, every person in the customers will contribute 0.11 HK dollars per live streaming. You find that the income of the webcasters may be dropped even they have more customers than before by this result. It is hard for the webcasters to attract customers to watch their live streaming and you also don't want the webcasters to waste their effort. Could you please give suggestions for the ranges for customer numbers to the webcasters according to this big data analysis so that the webcasters can avoid the situation that they have more customers but less income?