ASSIGNMENT 1

Software testing and reliability

HUU HUNG PHAM
102643407@student.swin.edu.au

Task 1: Explain and show all details on how to design test cases.

Program Under Test

Consider the following program:

```
Input A, B    // A and B are integer variables
A = (A – B)
C = A * B    // C is a real variable
Output C
```

Assume that there are only four feasible arithmetic operators, namely, + (addition), - (subtraction), * (multiplication) and / (division).

With the given program, I would like to use "Trial and Error" method to create a good test case.
Trial and error is the process of determining a good way to obtain the desired outcome by recognizing and removing bad test cases through trial.

First, we need to find possible alternatives of the given program.
The above program contains two arithmetic operators, namely, "-" in the statement of "A = (A – B)" and "*" in the statement of "C = A * B".
Incorrect use of arithmetic operators means using the wrong arithmetic operators in one, or both statements.
The incorrect arithmetic operators for statement A = A – B includes "+", "*", "/".
The incorrect arithmetic operators for statement C = A * B includes "-", "+", "/".
Combine the two statements, we get all 15 variants of the given program with wrong arithmetic including:

Given statement:	A = A - B	C = A * B
	-	/
	-	+
	-	-
	+	+
	+	-
	+	*
	+	/
	*	+
	*	-
	*	*
	*	/
	/	+
	/	-
	/	*
	/	/

Huu Hung Pham - 102643407

The result we have from the original arithmetic operators should be different to all variants in order to get a good test case.

Problem when designing the test case of this program:
A and B are intergers, but the first statement A = A – B has a variant which is A = A / B which can make A a real number. That is the reason why our good test case should have number A = n times B

Task 2: Suppose you use test case (A= 8, B=2) to test the above program. Is this test cases able to achieve the required testing objective? Provide your answer with justifications.

Using the table in task 1:
Try A=8 and B=2 in the table, we have:

Given statement:	A = A - B	6	C = A * B	12
1	-	6	/	3
2	-	6	+	8
3	-	6	-	4
4	+	10	+	12
5	+	10	-	8
6	+	10	*	20
7	+	10	/	5
8	*	16	+	18
9	*	16	-	14
10	*	16	*	32
11	*	16	/	8
12	/	4	+	6
13	/	4	-	2
14	/	4	*	8
15	/	4	/	2

Justification:
We can observe that in line 4 variant, the result is same with the original arithmetic operators which is 12. So this test case is not able to achieve testing objective.

Task 3: Based on your design in Task 1, what is (or are) the concrete test case (or cases) that can achieve the above testing objective?

Try A=10 and B=2 in the table, we have:

Given statement:	A = A - B	8	C = A * B	16
1	-	8	/	4
2	-	8	+	10
3	-	8	-	6

4	+	12	+	14
5	+	12	-	10
6	+	12	*	24
7	+	12	/	6
8	*	20	+	22
9	*	20	-	18
10	*	20	*	40
11	*	20	/	10
12	/	5	+	7
13	/	5	-	3
14	/	5	*	25
15	/	5	/	2,5

Justification:
This is a good test case base on my design in task 1. As all the result of variants are different from the original arithmetic operators. Thus, it can detect all the incorrect operators.
In addition, A = 5*B, so there would be no problem when we proceed with the variant A = A/B of statement 1.