

Example on Mutation Testing

Dr. Sangharatna Godbole
Assistant Professor
Dept. of CSE
NIT Warangal

Example

Consider the following program. Generate five mutants and design test cases taking into account each mutant, using mutation testing. Calculate mutation score of your test suite.

```
main(){
float x,y,z;
printf("Enter values of three variables x,y,z");
scanf("%f%f%f",&x,&y,&z);
if(x>y){
if(x>z)
printf("%d is greatest",x);
else
printf("%d is greatest",z);
}
else{
if(y>z)
printf("%d is greatest",y);
else
printf("%d is greatest",z);
}
}
```

Solution

The instrumented program is given below.

```
1  main(){
    float x,y,z;
2  printf("Enter values of three variables x,y,z");
3  scanf("%f%f%f",&x,&y,&z);
4  if(x>y){
5  if(x>z)
6  printf("%d is greatest",x);
    else
7  printf("%d is greatest",z);
    }
    else{
8  if(y>z)
9  printf("%d is greatest",y);
    else
10 printf("%d is greatest",z);
    }
11 }
```

Test suite

Consider the test suite given in Table 1.

Table 1: Test suite for Q ??

Sl. No.	x	y	z	Expected Output
1	6	10	2	10
2	10	6	2	10
3	6	2	10	10
4	6	10	20	20

Mutated Statements

Table 2: Mutated statements

Mutant No.	Line No.	Original line	Modified line
M1	4	<code>if(x > y)</code>	<code>if(x < y)</code>
M2	4	<code>if(x > y)</code>	<code>if(x > (y + z))</code>
M3	5	<code>if(x > z)</code>	<code>if(x < z)</code>
M4	8	<code>if(y > z)</code>	<code>if(y = z)</code>
M5	7	<code>printf("%d is greatest",z);</code>	<code>printf("%d is greatest",y);</code>

Actual Output of mutant M1

The mutated line numbers and changed lines are given in Table 2. The actual output obtained by executing the mutants M1-M5 is shown in Tables 3–7.

Table 3: Actual output of mutant M1

Test case	x	y	z	Expected output	Actual output
1	6	10	2	10	6
2	10	6	2	10	6
3	6	2	10	10	10
4	6	10	20	20	20

Actual output of mutant M2

Table 4: Actual output of mutant M2

Test case	x	y	z	Expected output	Actual output
1	6	10	2	10	10
2	10	6	2	10	10
3	6	2	10	10	10
4	6	10	20	20	20

Actual output of mutant M3

Table 5: Actual output of mutant M3

Test case	x	y	z	Expected output	Actual output
1	6	10	2	10	10
2	10	6	2	10	2
3	6	2	10	10	6
4	6	10	20	20	20

Actual output of mutant M4

Table 6: Actual output of mutant M4

Test case	x	y	z	Expected output	Actual output
1	6	10	2	10	10
2	10	6	2	10	10
3	6	10	2	10	10
4	6	10	20	20	10

Actual output of mutant M5

Table 7: Actual output of mutant M5

Test case	x	y	z	Expected output	Actual output
1	6	10	2	10	10
2	10	6	2	10	10
3	6	2	10	10	2
4	6	10	20	20	20

Additional test case

Table 8: Additional test case

Test case	x	y	z	Expected Output
5	10	5	6	10

Output of added test case

Table 9: Output of added test case

Test case	x	y	z	Expected output	Actual output
5	10	5	6	10	6

Table 10: Revised Test suite

Sl. No.	x	y	z	Expected Output
1	6	10	2	10
2	10	6	2	10
3	6	2	10	10
4	6	10	20	20
5	10	5	6	10

Mutation Score

Mutation score=Number of mutants killed/Total number of mutants=4/5=0.8

Higher the mutant score, better is the effectiveness of the test suite. The mutant M2 is live in the example. We may have to write a specific test case to kill this mutant. The additional test case is given in Table 8.

Now, when we execute test case 5, the actual output will be different from the expected output (Table 9), hence the mutant will be killed. This test case is very important and should be added to the given test suite. Therefore, the revised test suite is given in Table 10.