

Teste, Verificação e Validação de Software

Mutation Testing
with



pitest.org

Exercício Prático

1 GDC

- a) Através da ferramenta *Pitclipse* identifique os mutantes presentes na função **GDC** da classe **Algorithm**.

```
public int gcd(int x, int y) {  
    int tmp;  
    while (y != 0) {  
        tmp = x % y;  
        x = y;  
        y = tmp;  
    }  
    return x;  
}
```

Figura 1 - Função gcd

- b) Encontre a melhor solução com o método de eliminação de mutantes.
- c) Teste a sua solução com a ferramenta *Pitclipse* e verifique se é possível eliminar os mutantes.

2 Min

- a) Através da ferramenta *Pitclipse* identifique os mutantes presentes na função **Min** da classe **Algorithm**.

```
public int Min(int x, int y) {  
    int v;  
  
    if (x < y)  
        v = x;  
    else  
        v = y;  
  
    return v;  
}
```

Figura 2 - Função min

- b) Encontre a melhor solução com o método de eliminação de mutantes e verifique com a ferramenta *Pitclipse* se é possível eliminar os mutantes.
- c) O que conclui dos mutantes da função **Min** da classe **Algorithm**?

3 NumZero e NegateArray

a) Através da ferramenta *Pitclipse* identifique os mutantes presentes na função **NumZero** e **NegateArray** da classe **Algorithm** e encontre possíveis soluções de teste capazes de os eliminar.

```
public int numZero(int[] x) {
    int count = 0;

    for (int i = 0; i < x.length; i++)
        if (x[i] == 0)
            count++;

    return count;
}

public void negateArray(final float i, float a[]) {
    for(int k = 0; k < a.length; k++)
        a[k] = a[k] * (-i);
}
```

Figura 3 - Função numZero e negateArray

4 IsLessThanThree

a) A função **IsLessThanThree** já tem um teste associado, porém mesmo com um **assert** aplicado e estando correto, os mutantes não morreram.

Por favor corrija a situação.

```
public boolean isLessThanThree(int number) {
    return (number < 3);
}
```

Figura 4 - Função isLessThanThree

```
@Test
public void testLessThanThree() {
    assertTrue(this.alg.isLessThanThree(2));
}
```

Figura 5 - Teste da função

Mutantes Vivos

```
changed conditional boundary
replaced return of integer sized value with (x == 0 ? 1 : 0)
```

Bom trabalho!