

Práctica #5: Search-Based Testing

1) Dado el siguiente programa:

```
static void foo(int x){  
L1:  if (x>0) {  
L2:    while (x!=0) {  
L3:      x=x-1;  
L4:      if (x==0) {  
L5:        System.out.println("Gol!");  
      }  
    }  
  }  
L6:  return;  
}
```

a) Escribir el Control-Flow Graph (CFG)

b) Calcular los dominadores de cada nodo

c) Calcular los post-dominadores de cada nodo

e) Escribir el Control-Dependency Graph (CDG)

2) Dado el siguiente programa:

```
static int sum(int x, int y) {  
L1:  if (x==0) {  
L2:    return x;  
L3:  } else if (y==0) {  
L4:    return y;  
L5:  } else {  
L6:    return x+y;  
  }  
}
```

a) Escribir el Control-Flow Graph (CFG)

b) Calcular los dominadores de cada nodo

c) Calcular los post-dominadores de cada nodo

e) Escribir el Control-Dependency Graph (CDG)

3) Dado el siguiente programa:

```
public int M(int[] b, int n){  
L1:  int s = 0;  
L2:  int i = 0;  
L3:  while (i != n) {  
L4:    s = s + b[i];  
L5:    i = i + 1;  
L6:  }  
L7:  return s;  
}
```

Generación Automática de Casos de Test - 2018

Ejercicios

- a) Escribir el Control-Flow Graph (CFG)
- b) Calcular los dominadores de cada nodo
- c) Calcular los post-dominadores de cada nodo
- e) Escribir el Control-Dependency Graph (CDG)