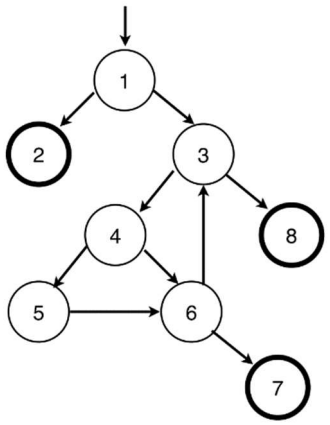
UNIVERSIDADE DA BEIRA INTERIOR
Faculdade de Engenharia | Departamento de Informática

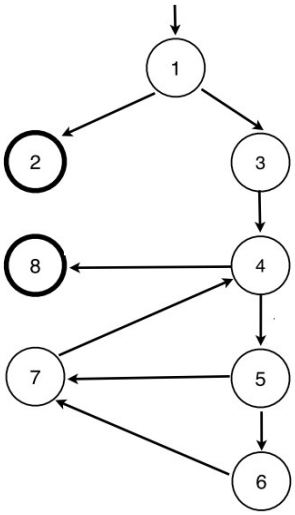
BADGE#05

Qualidade de Software - 2019/20

1. Tendo em conta o grafo à direita identifique:
   1.1. TR(NC)
   1.2. TR(EC)
   1.3. Identifique *test paths* que satisfaçam NC e não EC
   1.4. Identifique *test paths* que satisfaçam EC
   1.5. Será possível satisfazer EC apenas com *test paths* que não tenham ciclos?



2. Tendo em conta o grafo à direita identifique:
   1.1. TR(NC)
   1.2. TR(EC)
   1.3. TR(EPC)
   1.3. Identifique *test paths* que satisfaçam NC e não EC
   1.4. Identifique *test paths* que satisfaçam EC



3. Tendo em conta o seguinte código fonte:
   3.1. Desenhe o CFG da funçao *isPalindrome*
   3.2. Identifique TR(NC), TR(EC), TR(EPC), TR(PPC)
   3.3. Identifique, se possivel, casos de teste que satisfaçam:
      3.1.1. NC mas não EC
      3.1.2. EC mas não EPC
      3.1.2. EPC

```
public static boolean isPalindrome(String s) {
  if (s == null)
    throw new NullPointerException();
  int     left = 0;
  int     right = s.length() - 1;
  boolean result = true;
  while (left < right && result == true) {
    if (s.charAt(left) != s.charAt(right)) {
      result = false;
    }
    left++;
    right--;
  }
  return result;
}
```

4. Tendo em conta o seguinte código fonte:
   4.1. Desenhe o CFG da função *max*
   4.2. Determine o diversos *def/uses*
   4.3. Determine os *du-paths*

```java
public static int max(int[] v) {
  if (v == null || v.length == 0)
    throw new IllegalArgumentException();
  int max = v[0];
  int n = v.length;
  for (int i = 1; i < n; i++) {
    if (v[i] > max) {
      max = v[i];
    }
  }
  return max;
}
```

5. Para cada um dos grafos responda ás seguintes questões:

Graph I.
$N = \{0, 1, 2, 3, 4, 5, 6, 7\}$
$N_0 = \{0\}$
$N_f = \{7\}$
$E = \{(0,1), (1,2), (1,7), (2,3), (2,4), (3,2), (4,5), (4,6), (5,6), (6,1)\}$
$def(0) = def(3) = use(5) = use(7) = \{x\}$

Test Paths:
$t1 = [0, 1, 7]$
$t2 = [0, 1, 2, 4, 6, 1, 7]$
$t3 = [0, 1, 2, 4, 5, 6, 1, 7]$
$t4 = [0, 1, 2, 3, 2, 4, 6, 1, 7]$
$t5 = [0, 1, 2, 3, 2, 3, 2, 4, 5, 6, 1, 7]$
$t6 = [0, 1, 2, 3, 2, 4, 6, 1, 2, 4, 5, 6, 1, 7]$

Graph II.
$N = \{1, 2, 3, 4, 5, 6\}$
$N_0 = \{1\}$
$N_f = \{6\}$
$E = \{(1,2), (2,3), (2,6), (3,4), (3,5), (4,5), (5,2)\}$
$def(x) = \{1, 3\}$
$use(x) = \{3, 6\}$ // Assume the use of x in 3 precedes the def

Test Paths:
$t1 = [1, 2, 6]$
$t2 = [1, 2, 3, 4, 5, 2, 3, 5, 2, 6]$
$t3 = [1, 2, 3, 5, 2, 3, 4, 5, 2, 6]$
$t4 = [1, 2, 3, 5, 2, 6]$

   5.1. Desenhe o respetivo grafo
   5.2. Identifique todos os *du-paths* referentes a *x*.
   5.3. Identifique o *minimal test path* que satisfaça todos os *defs* coverage referentes a *x*.
   5.4. Identifique *test paths* que satisfaçam todas os *uses coverage* referentes a *x*.