Universiteti Politeknik i Tiranës

Fakulteti i Teknologjisë së Informacionit

Dega: Inxhinieri Informatike

Grupi: III-B

Viti akademik 2024-2025

Punë laboratori nr. 4

Lënda: Algoritmike dhe programim i avancuar

Punoi: Piro Gjidhima   Pranoi:Msc Alba Haveriku

Laborator 4

Ushtrim 3

Odd length directed cycle. Design a linear-time algorithm to determine whether a digraph has an odd-length directed cycle.

Klasa OddLengthDirectedCycle

```java
package Laboratore.Lab4;

import edu.princeton.cs.algs4.Digraph;

public class OddLengthDirectedCycle {
    private final boolean[] marked;
    private final int[] vertex;
    private boolean hasOddCycle;

    public OddLengthDirectedCycle(Digraph G) {
        marked = new boolean[G.V()];
        vertex = new int[G.V()];
        hasOddCycle = false;

        for (int v = 0; v < G.V(); v++) {
            if (!marked[v]) {
                dfs(G, v, 0);
            }
        }
    }

    private void dfs(Digraph G, int v, int c) {
        marked[v] = true;
        vertex[v] = c;

        for (int w : G.adj(v)) {
            if (hasOddCycle) return;

            if (!marked[w]) {
                dfs(G, w, 1 - c);
            } else if (vertex[w] == vertex[v]) {
                hasOddCycle = true;
                return;
            }
        }
    }

    public boolean hasOddCycle() {
        return hasOddCycle;
```
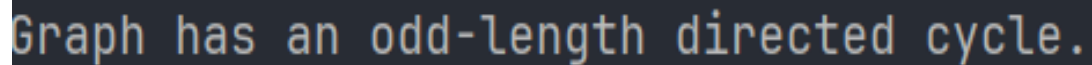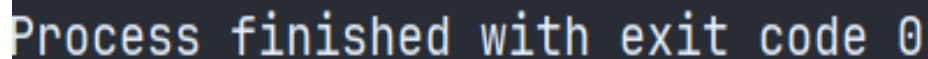
```
    }
}
```

Klasa Lab4

package Laboratore.Lab4;

import edu.princeton.cs.algs4.Digraph;

public class Lab4 {

   public static void main(String[] args) {

      Digraph G = *new* Digraph(4);
      G.addEdge(0, 1);
      G.addEdge(1, 2);
      G.addEdge(2, 0);
      G.addEdge(2, 3);

      OddLengthDirectedCycle finder = *new* OddLengthDirectedCycle(G);
      if (finder.hasOddCycle()) {
         System.out.println("Graph has an odd-length directed cycle.");
      } *else* {
         System.out.println("Graph does not have an odd-length directed cycle.");
      }
   }
}

```
Graph has an odd-length directed cycle.


Process finished with exit code 0
```