

# ALGORYTMY I STRUKTURY DANYCH

## GRAFY

### Zadanie:

Zaimplementuj graf w postaci:

- (i) macierzy sąsiedztwa,
- (ii) uporządkowanych list sąsiadów.

Dla obu postaci grafów zdefiniuj zestaw następujących funkcji:

**init**( $n$ ) Inicjalizacja grafu pustego z podaniem liczby wierzchołków.

**insert**( $u, v$ ) Dodanie do grafu krawędzi ( $u, v$ ) (jeśli nie ma jej w grafie to funkcja dodaje krawędź i zwraca wartość 1, w przeciwnym przypadku funkcja zwraca wartość 0).

**delete**( $u, v$ ) Usunięcie z grafu krawędzi ( $u, v$ ) (jeśli jest w grafie to funkcja usuwa krawędź i zwraca wartość 1, w przeciwnym przypadku funkcja zwraca wartość 0).

**search**( $u, v$ ) Sprawdzenie, czy w grafie jest krawędź ( $u, v$ ) (funkcja odpowiada wartością 0, gdy krawędzi nie ma, albo 1, gdy krawędź jest).

**degree**( $v$ ) Sprawdzenie stopnia wierzchołka  $v$  (funkcja odpowiada ile sąsiadów ma  $v$ ).

Sprawdź, jak szybkie są obie implementacje grafu dla zadanych danych.

### Dane:

W pierwszym wierszu z danymi jest podana liczba  $n$  ( $1 \leq n \leq 1000$ ) oznaczająca liczbę wierzchołków w grafie (wierzchołki są numerowane od 0 do  $n - 1$ ).

W drugim wierszu jest podana liczba  $k$  ( $1 \leq k \leq 1000000$ ) oznaczająca liczbę operacji jakie trzeba będzie wykonać na początkowo pustym grafie  $n$ -wierzchołkowym.

W kolejnych  $k$  wierszach są zapisane operacje do wykonania. Każda operacja ma postać litery (symbol funkcji wykonywanej na grafie) i oddzielonych pojedynczą spacją liczb (numery wierzchołków):

- I  $u v$  dodanie do grafu krawędzi ( $u, v$ )
- D  $u v$  usunięcie z grafu krawędzi ( $u, v$ )
- S  $u v$  sprawdzenie, czy w grafie jest krawędź ( $u, v$ )
- G  $v$  sprawdzenie stopnia wierzchołka  $v$

**Wyniki:**

W wyniku należy wypisać  $k$  wierszy, w których będą zapisane wyniki (liczby) kolejno wykonywanych na grafie operacji.

**Sprawozdanie:**

W sprawozdaniu należy zamieścić opis przeprowadzonych testów. Podziel testy na grupy związane z wykonywanymi operacjami: (I D S), (I D G) oraz (I D S G). Napisz, jakie czasy otrzymałeś/otrzymałaś testując obie implementacje grafów.