Homework Assignment 1

Due: Friday, November 13

Q1: **Array initialization**

Describe a data structure supporting the following operations in worst-case constant time:

- $\mathsf{Init}(n)$ – create a "list" L of length n with all entries set to nil.
- $\mathsf{Set}(i, v)$ – set the ith entry of L to v.
- $\mathsf{Get}(i)$ – return the ith entry of L.

The operations may use a procedure $\mathsf{alloc}(n)$ that in worst-case constant time returns an uninitialized array of length n, *i.e.*, all entries of the allocated array contain random information.

Q2: **Planar graph representation**

Show how to represent a planar graph $G = (V, E)$ in $O(|V|)$ space such that "$(u, v) \in E$?" queries can be answered in constant time (the solution should *not* use hashing).

Q3: **Sink**

Given the adjacency-matrix of a directed graph $G = (V, E)$, show how to compute in $O(|V|)$ time if G has a *sink*, *i.e.*, a node of in-degree $|V| - 1$ and out-degree 0.

Q4: **Scorpion** *

A *scorpion* is an undirected graph $G = (V, E)$ of the following kind: there are three special vertices called the *sting*, the *tail* and the *body* of degree 1, 2 and $|V| - 2$, respectively. The sting is connected only to the tail; the tail is connected only to the sting and the body; and the body is connected to all vertices except the sting. Besides being connected to the body, the remaining $|V| - 3$ vertices may be connected to each other arbitrarily.

Give an algorithm that makes only $O(|V|)$ probes of the adjacency matrix of G and determines whether G is a scorpion.