York University | EECS 4101/5101 | November 25, 2022

Homework Assignment #8
Due: December 2, 2022 at 11:59 p.m.

[3] **1.** Consider the following algorithm to check whether there are any duplicates in an unsorted array A of n elements, each drawn from the universe $\{1, 2, \ldots, N\}$. Assume p is a prime number with $p > N$. The algorithm uses chaining on a hash table of size n.

```
choose a random a ∈ {1, ..., p − 1} and b ∈ {0, ..., p − 1}
let h be the hash function defined by h(x) = ((ax + b) mod p) mod n
let B[0..n − 1] be an array of linked lists; initially all lists are empty
for j ← 0..n − 1
    iterate across the linked list B[h(A[j])] looking for the element A[j]
    if found, stop and return "Duplicate found"
    else append A[j] to the list B[h(A[j])]
    end if
end for
return "No duplicates found."
```

What is the expected running time of this algorithm? Express your answer using big-O notation, and justify your answer. Your bound should hold for every possible input array A.

[4] **2.** Consider directed graphs on n nodes, labelled $1, 2, 3, \ldots, n$. Such a graph is called *sparse* if the number of edges is much smaller than n^2. Such a graph can be compactly represented as an (unsorted) list of its edges. Each element in this list is a pair (i, j), indicating that there is an edge from node i to node j.

Given the list representation of two directed graphs on n nodes, each containing at most m edges, give a randomized algorithm to determine whether the first is a subgraph of the second. The expected running time of your algorithm should be $O(m)$, and the amount of space used should be $O(m)$ in the worst case.

You may assume that you are given a prime number $p \geq n^2$.