University of Puerto Rico at Rio Piedras
Department of Mathematics
Qualifying Exam
Algorithms (Data Structures II)

February 4, 2020

Your Name	Your Student ID

Instructions: Solve **exactly three** out of the following **five** exercises. Please read carefully, write clearly and show all your work to get credit. Good luck!

Exercise 1 (100 points)

(a) (50 points) For each of following pairs of functions, state which one grows faster and prove your results.

(ii) $n^2/\log n$ or $\log^2 n$

(iii) $n^{1/n}$ or $\log n$

(b) (50 points) In each of the following cases, find asymptotic upper and lower bounds for $T(n)$. Make your bounds as tight as possible, and justify your answers.

(i) $T(n) = 2T(\frac{n}{2}) + n^4$. You may assume that n is a power of 2 and that $T(1)$ is constant.

(ii) $T(n) = 3T(\frac{n}{3}) + n$. You may assume that n is a power of 3 and that $T(1)$ is constant.

Exercise 2 (100 points) [Maximum Product Subarray]
The MaxProdSubarray problem has the following specifications:

Input: An array A containing $n \geq 1$ positive and negative integers but not zeroes.
Output: Two indices i, j, such that the following product is the maximum possible:

$$A[i] * A[i+1] * \cdots * A[j]$$

Give pseudocode for this problem. What is the running time of your algorithm?

Exercise 3 (100 points)
You have a mixed pile of n nuts and n bolts and need to quickly find the corresponding pairs of nuts and bolts. Each nut matches exactly one bolt, and each bolt matches exactly one nut. By fitting a bolt and a nut together, you can see which is bigger, but it is not possible to directly compare two nuts or two bolts. Give an efficient algorithm for solving this problem.

Exercise 4 (100 points) [Overlapping Intervals]
Given a set of closed intervals $\{[X_1, Y_1], [X_2, Y_2], \ldots, [X_n, Y_n]\}$, write an algorithm to print all non-overlapping intervals after merging overlapping intervals. What is the running time of your algorithm?

For example,
Input: $\{[2,3],[4,6],[7,8],[1,5],[8,10],[12,15]\}$
Output: $\{[1,6],[7,10],[12,15]\}$.

Exercise 5 (100 points) [Maze Problem]
A maze is constituted of a $n \times n$ grid W with coordinates expressed as pairs (x, y), called cells, where $x = 1 \ldots n$, $y = 1 \ldots n$. The cell (n, n) is the *exit* of the maze. The cell $(1, 1)$ is its *entrance*. We can move between adjacent cells only, in directions *top*, $left$, *right*, *bottom*. Some cells may be occupied by a *wall*, which prevent passing through them. A cell with a wall is identified by $W(x, y) = 1$. Cells that can be traversed are identified by $W(x, y) = 0$. We assume entrance and exits are free of walls: $W(1, 1) = W(n, n) = 0$. A path is valid if it is a sequence of adjacent cells that does not include cells with walls. Design an algorithm to output a *shortest valid path* from the entrance to the exit:

Input: n, W
Output: Path as a list $[(1, 1), (x_1, y_1), \ldots, (n, n)]$of cells that starts with $(1, 1)$ and ends with (n, n). The output should be the empty path [] in case no such path exists.

(a) Propose an algorithm to compute such a path if it exists. (partial credit for detecting existence or returning a valid path that is not the shortest)

(b) Provide upper and lower bounds of its complexity as a function of n. Try to provide tight bounds and justify them.