University of Puerto Rico at Rio Piedras
Department of Mathematics
Qualifier: Algorithms

September 14, 2017

Your Name	Your Student ID

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

Exercise 1 (100 points)
Part A (30 points) Imagine a group of n participants in the first round of a chess tournament. Suppose n is an even number. The tournament committee wants to set up a pairing for the first round. Prove by induction that the number of pairings among which the committee has to choose one, is equal to the product of the odd numbers from 1 to n, i.e.

$$(n-1) * (n-3) * (n-5) * \cdots * 3 * 1$$

For example, assuming we have $n = 4$ persons, A, B, C, D, there are 3*1 pairings:
Pairing 1: {(A,B),(C,D)}
Pairing 2: {(A,C), (B,D)}
Pairing 3: {(A,D), (B,C)}

Part B (70 points) Write pseudo-code that takes as input an even number n and prints out all possible pairings of n items (for the pairings you can use letters as above, or numbers). Hint: you can try recursion.

Exercise 2 (100 points)
A **palindrome** is a nonempty string over some alphabet that reads the same forward and backward. Examples of palindromes are all strings of length 1, *civic, anilina*, and *racecar*. Give an algorithm to find the longest palindrome that is a substring of a given input string. For example, given the input *bananas*, your algorithm should output *anana*. What is the running time of your algorithm?

Exercise 3 (100 points)
The **0-1 knapsack problem** is the problem of finding, given an integer $W \geq 1$, items $1, \ldots, n$, and their values, $v_1, \ldots, v_n$, and their weights, $w_1, \ldots, w_n$, a selection, $I \subseteq \{1, ..., n\}$, that maximizes $\sum_{i \in I} v_i$ under the constraint $\sum_{i \in I} w_i \leq W$.
An example: Pedro is going to the island of Culebra. He is allowed to carry only one bag with him and the bag holds no more than 16 pounds, so he can't bring all what he wants. Thus, he weighted and valued each item he wants to bring. What should he be putting in the bag?

Item	**Weight (lb)**	**Value**
TV	8	20
Telescope	10	25
Chessboard	2	8
SW radio	4	12
Harmonica	1	5
Tablet	4	6
Inflatable Life-Size R2D2 Doll	1	8

Propose a greedy strategy and show that it does not necessarily yield an optimal solution for the 0–1 knapsack problem.

Exercise 4 (100 points)
Give a divide-and-conquer algorithm that runs in time $O(\log n)$ that, given a sorted array of distinct integers $A[1, \ldots, n]$, determines whether there is an index i for which $A[i] = i$ or not.

Exercise 5 (100 points)
Give pseudo-code for an algorithm that takes as input a sorted array A of size n and determines if there is an element that appears more than $n/2$

times in A. The algorithm should run in $O(\log n)$ time. Give an argument about the correctness and running time of your algorithm.