Universidad del Rosario
School of Engineering, Science and Technology
Applied Math and Computer Science
Operating Systems

Exam I

1. Apply the FCFS Non-preemptive, SJF preemptive and Round Robin CPU scheduling algorithms to the following list of processes. Assume *q*=4, and assume a single tie breaker rule: Pi > Pj, i < j: (1.5 points)

Process	Time of arrival	CPU	I/O	CPU
P1	0	5	4	3
P2	3	3	5	6
P3	6	7	3	5
P4	8	4	3	7

Calculate the following performance indicators (0.5 points):
- Average waiting time
- CPU Utilization
- Throughput
- Number of context switch

According to the previous results and, assuming that a context switch takes 3 cycles, discuss: (0.5 points):
- Which of the algorithms had the best real performance?
- If you could make a change to the policies of the scheduling algorithms, which would it be in order to improve their performance?

2. Define what a Process Control Block is and what information does it store. Describe the structure for a modern operating system, based on on-line sources (the book material is not allowed to be cited) (Windows 8 or later, Mac OS, Linux, Unix, Solaris, etc.) Include references. (0.5 points)

3. Describe with your own words the Process Scheduling algorithm used in any modern operating system, based on on-line sources (the book material is not allowed to be cited) (Windows 8 or later, Mac OS, Linux, Unix, Solaris, etc.) Include references. (0.5 points)

4. Build an algorithm that, using a process creation tree technique, allows you to play the "Cold-Hot" game on a set of semi-ordered data array (they are not sequential numbers, each number is at most 10 spaces from its ideal position), according to the following rules (1.5 points):

    a. The process P0 has stored in a variable the number that is being searched for and has an array of ten thousand numbers. In addition, it has a method that compares a number X

with the secret number and returns "Freezing cold" if it is more than 100 positions away, "Cold" if it is more than 30 positions away, and "Hot" if the difference is 30 or less.

b. Each child process is responsible for searching a range of the search structure. If a node's range is less than 100 and its upper and lower bounds are “Freezing cold”, the process should stop searching. If a node's range is less than 30 and its upper and lower limits are "Cold", the process should stop searching. If the range is less than or equal to 30 and in any of the limits it is "Hot", a sequential search can be performed in this interval or create more children, if the range is larger than 30.
c. The final tree must not have more than 10 levels. A process cannot have more than 5 children processes. The process can have an array of PIDs variables.
d. The process that finds the element writes on the screen the position of the vector in which it was found.

Include the following items:

a. Description of solution
b. Submit the solution code
c. Present an example of the execution of the algorithm, using the process tree representation. Include the values of the variables of interest.