CSCE-4600 Operating Systems Design

Homework #2
Due 2-14-2019 11:59pm
Submission per Canvas

1. (20pts) Consider a system of 9 processes, **P** = {p1, …, p9}
   Associated with the system are 6 memory cells, **M** = {M1, .., M6}

   The domain and range for each process is given in the following table:

Process pi	Domain D(pi)	Range R(pi)
p1	M1, M2	M3
p2	M1	M5
p3	M3, M4	M1
p4	M3, M4	M5
p5	M3	M4
p6	M4	M4
p7	M5	M5
p8	M3, M4	M2
p9	M5, M6	M6

   In addition, you are given the following precedence relation:
   ➔ = {(1,2),(1,6),(2,3),(2,4),(2,5),(3,6),(3,8),(4,6),(4,7),(5,7),(5,8),(6,8),(6,9),(7,9),(8,9)}

a. (10pts) Construct the Precedence Graph (not containing any redundant edges)

b. (10pts) Determine if the system above is always determinate. If it is not, add to ➔ necessary elements to make it determinate.

2. (30pts) Write a simple sequence-number system through which three concurrent processes, P1, P2, and P3, **each obtain unique integers** in the range [1, 500]. Use the fork() call to create P, P2, and P3. Given a file, F, containing a single number, each process must perform the following steps:

   a. Open F
   b. Read the sequence number N from the file
   c. Close F
   d. Output N and the process' PID (either on screen or test file)
   e. Increment N by 1
   f. Open F
   g. Write N to F
   h. Close F.

   Describe the behavior of your program and explore the reason for this behavior. Provide evidence for your conclusion in form of test-output. You must clearly document your code.

NOTE: All programs must compile and execute on the CSE machines. It is imperative **that the sequence number files is located on the local disk. On Linux, the \tmp directory is located on the local file system.**

3. (30 pts) Write a program in C or C++, which simulates the generation of a set of k processes. Each process represented by a 3-tupel containing a unique process PID, the number of CPU-cycles required to complete the process, and the size of the memory footprint. The required number of cycles is chosen from the interval <1,000,11,000> with a mean of 6,000. While it is acceptable to distribute the required cycles uniformly, (I suggest that you attempt to implement a different distribution.) The memory footprints of processes fall in the range of 1KB to 100KB with a mean memory footprint of 20 KB. You need to represent the set of k processes with a data structure of your choice. Show how the values (required cycles and memory footprint) are distributed over your set of processes. You must submit your program, and a short description of your approach and the data structures used.

4. (20) Extend Peterson's SW-based MUTEX solution to work with *n* processes.