SOLUTION ASSIGNMENT 4 CMPT300 SUMMER 2015

1. **[75 points]** Consider Smaug's world as described in assignment 2. Consider only Smaug, treasure hunters and thieves. Smaug is a process, each treasure hunter is a process and each thief is a process. Use Posix semaphores (you probably want named semaphores) and shared memory (for counter and unnamed semaphores).

Assume each of the following

- The first process in the simulation will spawn Smaug.
- Each thief will be a separate process. Each treasure hunter will be a separate process
- Thieves arrive (new thief processes are created) at random times. The length of the interval between the arrivals of successive thieves is drawn from a uniform distribution. The length of interval n is the length of time between the arrival of thief n and the arrival of thief n+1.The distribution of possible interval lengths is uniform between 0 seconds and maximumThiefInterval seconds (0 probability of an arrival interval outside this range).
- Similarly treasure hunters arrive at random intervals drawn from a uniform distribution. The distribution of possible interval lengths is uniform between 0 seconds and maximumHunterInterval seconds (0 probability of an arrival interval outside this range).
- Smaug's treasure initially contains 30 jewels
- The probability that a particular treasure hunter or thief will not be defeated is winProb.
- Your program should request the values of maximumHunterInterval, maximumThiefInterval and winProb from the user and used those values to initialize your simulation.

Your simulation should terminate when any one of the following conditions is me

Smaug has defeated 4 treasure hunters or 3 thieves
Smaug has no treasure
Smaug has 80 jewels.

When your simulation terminates you must assure that no zombie (defunct) processes remain. At no time during execution should you have more than 10 zombies. You must also assure that all semaphores and mutexes allocated are released before your simulation terminates.

Your program should produce output (to the screen) each time something happens. That is when you run your program it should produce output that includes the following messages (in the order they actually happen). Note comments in brackets are not part of the message to be printed.

Treasure hunter PIDVal wandering in the valley
Thief PIDVal wandering in the valley
Treasure hunter PIDVal is travelling to the valley
Thief PIDVal is travelling to the valley
Thief PIDVal is playing with Smaug
Treasure hunter PIDVal is fighting Smaug
Treasure hunter PIDVal has been defeated and pays the price
Treasure hunter PIDVal has won and receives treasure
Thief hunter PIDVal has been defeated and pays the price

Thief hunter PIDVal has won and receives treasure
Thief PIDVal leaves
Treasure hunter PIDVal leaves
Smaug is going to sleep
Smaug has been woken up
~~Smaug is taking a nap~~
Smaug takes a deep breath
Smaug smells a thief
Smaug smells a treasure hunter
Smaug is fighting treasure a hunter
Smaug is playing with a thief
Smaug has defeated a thief
Smaug has been defeated by a thief
Smaug has added to his treasure he now has M jewels
Smaug has lost some treasure he now has M jewels
Smaug has finished a battle
Smaug has finished a game
Smaug has no more treasure

You may add additional messages if you wish. Replace PIDVal with the process ID of the thief or treasure hunter.

2. Consider a hard disk. The disk has N=128 tracks and 24 sectors per track. The seek time per track is 1 ms. The rotation speed of the disk is 7200rpm.

   a) **[7 points]** Assume that once the head is in position above the appropriate track you need to wait an average of half a rotation of the disk before the head can read your data. What are the maximum and minimum rotational latency of this disk? What is the average rotational latency of this disk?
   b) The following sequence of disk access requests have been received by the computer systems. The numbers are the track numbers. Numbering starts at 0 (the centremost track on the disk).
   45, 74, 52, 121, 11, 44, 2, 99, 76
   Determine the actual average seek length for this particular series of accesses based on each of the following algorithms. Assume the read head starts at track 48
      - **[6 points]** Shortest seek first (move the fewest number of tracks to the next entry)
      - **[6 points]** Elevator algorithm (towards the outside of the disk first, direction of increasing track number)
      - **[6 points]** Circular LOOK algorithm (towards the outside of the disk first, direction of increasing track number