**COMP 2604 Assignment 2 (8%)** **Due: 16th March 2018 @ 4.00pm**

**Question 1**

Write a complete C program using the `pthread_create()` system call that sorts a given *n*-element array of integers, where *n* is a positive integer supplied by the user. You can use a sorting algorithm of your choice (e.g., bubble sort, insertion sort, or selection sort). The array will be sorted by the child thread and the sorted array will be output by the parent thread. Have the parent thread invoke the `pthread_join()` system call to wait for the child thread to finish the sorting task. Perform necessary error checking to ensure that a positive integer *n* is passed on the command line.

**Question 2a, b, c**

If FIFO (First-In-First-Out) page replacement is used with four page frames and eight pages, how many page faults will occur with the address stream 0 1 7 2 3 2 7 1 0 3 if the four frames are initially empty? Draw a diagram showing the frame allocations for the address stream using FIFO. Now repeat this problem for the LRU (Least-Recently-Used) page replacement algorithm.

**b.** Consider a simple paging system using 16-bit logical and physical addresses and the page size is 1024 bytes. The process page table is given below.

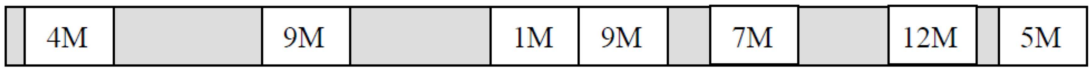
| | |
|---|---|
| 0 | 000101 |
| 1 | 000110 |
| 2 | 011001 |

What is the physical address of the logical address 0000010111011110? [2 marks]

**c.** What is the difference between internal and external fragmentation? [2 marks]

**Question 3**

Consider a swapping system in which memory consists of the following hole sizes in memory order: 4 MB, 9MB, 1 MB, 9 MB, 7 MB, 12 MB, and 5 MB.




Addresses go from left to right, the gray areas indicate blocks occupied by processes, and the white areas indicate free memory blocks.

(a) Draw diagrams illustrating the final memory configurations, clearly identifying the holes that are taken and created, for successive memory requests of 3 MB, 6MB, 1MB for the following memory management algorithms under dynamic partitioning:
   (i) First fit (FF)
   (ii) Best fit (BF)
   (iii) Worst fit (WF)
   (iv) Next fit (NF)

**Question 3 continued**

(b) A virtual memory system uses a page size of 4096 bytes. A virtual address consists of 32 bits.

(i) How many entries would a page table in this system contain? [1]

(ii) Explain carefully how the memory management unit converts a virtual address into an address in real memory. [6]

(iii) The system uses the "Not Frequently Used" page replacement algorithm. Describe the operation of this algorithm. [5]

(c) In a virtual memory management system, under what conditions are inverted page tables needed? How is such a table used? [4]

**Question 4**

a) What are the operations that can be performed on a semaphore? Describe each operation.

b) In systems that have processes with multiple threads, thread scheduling differs depending on whether user-level threads or kernel-level threads are used. Distinguish between these two approaches with the use of an example. Give an advantage of each approach. [5]

c) Write assembly pseudocode to illustrate how the TSL instruction can be used to achieve mutual exclusion. Briefly explain your code. [4]

d) (i) What is the main defect of both Peterson's solution and the solution given above using the TSL instruction? [1]

**Please ignore the marks listed next to the questions.**

Submission Instructions:

- Work in groups of 5
  - form your own groups and submit one assignment per group
  - Ensure each student name and ID is clearly stated on a cover page
- Implement your program for question 1 on a UNIX like platform such as Ubuntu Linux.
- The source code must be appropriately commented and structured to allow users to understand your code
- Upload a zipped file containing your solutions to myelearning before the deadline.
- Absolutely no late submissions