CMPUT 379 Final Exam [Harms]
April 25, 2001
Closed Book

Comments:

- This exam is worth 38% of your final grade. There are 5 questions and 4 pages. The mark distribution is given beside the questions. The total number of marks is 80.
- This is closed book exam. Calculators are not necessary nor allowed.
- If you are concerned about an interpretation of an exam question,ask me or state your assumptions and then answer the question.
- Be sure to show your work! Good Luck!

(15 marks) Question 1.

Give concise discussions/explanations for each the following:

a) Discuss the effect of quantum size on the performance of Round Robin CPU scheduling. Consider very small, medium range and very large quantum sizes.

b) Compare prevention and detection-and-recovery mechanisms for dealing with deadlock in an operating system.

c) Compare the performance of Shortest Seek Time First (SSTF) and SCAN disk scheduling algorithms.

d) Compare acyclic graph directory structures and tree directory structures.

e) Consider the following process behaviour. A particular reference string consists of 52 pages when the page size is 512 bytes; however, if the page size is changed to 1024, the number of pages in the reference string is 34. Explain,why doubling the page size does not result in halving the number of pages.

f) Compare authentication schemes where something is known by the user (e.g. a password) with schemes where something is known about the user (e.g. fingerprint, voice).

(10 marks) Question 2.

Consider the outlines of 3 processes given below. **a**, **b** and **c** are binary semaphores and **s1**, **s2** and **s3** are program statements. The semaphores are all initialized to 1. The processes are started concurrently.

```
Process1()    {             Process2()    {             Process2()    {
  while (1)   {               while (1)   {             while (1)    {
       ...                       ...                       ...
       P(c);                      P(b); P(c);               P(b); P(a); P(c);
       S1;                       S2;                        S3;
       V(b);                     V(b); V(c);               V(c); V(a);
       ...                    ...                        ...
}}                          }}                          }}
```

a) In what order are statements S1, S2, S3 executed?

b) Is deadlock possible? If so, explain how. If not, explain why not.

c) Suppose that it is necessary that S2 and S3 be executed mutually exclusively. What does this mean? Is mutual exclusion violation possible in the above scenario? Explain why or why not.

(10 marks) Question 3.

a) In a virtual memory management scheme that uses paging, if pages are of size 1024 bytes (2^{10} bytes), the virtual address is 32 bits and the physical address is 24 bits.

(i) What is the maximum number of entries that could be in a page table?

(ii) How many entries would be in an inverted page table?

b) Suppose that paging is used and the physical address is larger than the virtual address. Does this imply that there will be no page faults? Explain why or why not.

c) Suppose that a hybrid index block scheme is used for file allocation (similar to Unix). In this particular case assume that in the "inode" of the file there are 2 direct pointers (direct to data blocks), 1 pointer to a single level index block and 1 pointer to the first block of a two-level index block. What is the maximum size of file that can be created in this system if pointers are 4 bytes and the block size is 4096 bytes?

(25 marks) Question 5.

Consider the following3 file allocation strategies:

(i) File Allocation Table (FAT).

(ii) Contiguous File Allocation. First fit is used for allocating space.

(iii) Linked index block scheme

Answer the following questions for each scheme:

a) describe what is stored in main memory when a file is opened.

b) briefly describe a free block list organisation that works well with the particular storage allocation mechanism. Given your chosen free block mechanism and the particular storage allocation method, discuss how easy or hard it would be to add another block to the end of the file (an "append" operation)

c) discuss the storage overhead (including fragmentation) that can occur with the particular scheme.

d) discuss the performance in terms of access time and storage usage for accessing blocks in a very small file and in a very large file. Assume direct access is required rather than sequential.

e) discuss the effect of large versus small block size

(20 marks) Question 4.

Suppose that a process has the following reference string:

1 2 4 2 2 2 1 3 1 0 3 7 0 0 1 1 2 1 0 7 4 9 0 0 1 2

a) Suppose that local page replacement is used and the process is allowed 4 frames in a pure demand paging system. How many page faults would occur if the following page replacement algorithms are used.

(i) FIFO
(ii) LRU
(iii) Optimal

b) Another page replacement algorithm that people have proposed is called Least Frequently Used (LFU). In this algorithm, the page that has been referenced the least number of times since it was brought into main memory is victimised. Using the scenario described above, how many page faults would there be for LFU? Give a justification for this algorithm. What disadvantages does this algorithm have?

c) If the working set algorithm with window size 7 was applied to the reference string listed above, what are the working sets after the 6th, 12th, 18th and 24th references?

d) Define spatial locality and temporal locality. For each give an example of a program activity that could result in the locality.