Operating Systems

Examination Sheet

(14:50-16:20PM, February 3, 2010 – open book)

Student Name: _____ **ID:** _____

Problem 1 (20 points). Consider the two-dimensional array a like this

double a[150][150]

Where a[0][0] is at location 300, in a paged memory system with pages of size 300 words. Each word has 4 bytes in length. A small process is in page 0 (locations 0 to 299) for manipulating the matrix; thus, every instruction fetch will be from page 0. For three page frames, how many page faults are generated by the following array-initialization loops (C-language), using LRU replacement, and assuming page frame 1 has the process in it, and the other two are empty initially:

A).
```
for (i = 0; i < 150; i++)
  for (j = 0; j < 150; j++)
    a[i][j] = 0;
```
Answer: _____75_____

B).
```
for(j = 0; j < 150; j++)
  for(i = 0; i < 150; i++)
    a[i][j] = 0;
```
Answer: _____11250_____

Problem 2 (35 points). Consider a file currently consisting of **150** blocks numbered from 0 to 149. Assume that the file **control block** (and the index block, in case of indexed allocation) is already in memory. Calculate how many disk I/O operations are required for contiguous, linked, and indexed (single-level) allocation strategies, if, for one block, the following conditions hold. *In the contiguous-allocation case, assume that there is no room to grow in the end, but there is room to grow in the beginning.* Assume that the information block to be added is stored in memory. Fill the following table.

	contiguous	linked	indexed (single-level)
The block is added in the beginning.	1 W.D. = **1**	1 W.D. + 1 W.L. = **2**	1 W.D. = **1**
The block is added after the block 75	76*(1R.D. +1W.D.)+1W.D. = **153**	76 R.L., 1 W.D., 2 W.L. = **79**	1 W.D. = **1**
The block is added in the end.	150*(1R.D.+1W.D.)+1 W.D.=**301**	1W.D.+1W.L.+1W.L. = **3**	1 W.D. = **1**
The block is removed from beginning.	**0**	1W.L.=**1**	0
The block 35 is removed.	35*(1R.D.+1W.D.)=**70**	36R.L.+1W.L.=**37**	0
The block is removed from end.	**0**	150R.L.=**150**	0

(It is a bit ambiguous that the block 75 refers to the 75th block starting from 1 to 150, or the block with the index 75 starting from 0 to 149? Here, I assume that it refers to the block with 75.
W.D ~ Write Data; W.L. ~ Write Link; R.D. ~ Read Data; R.L. ~ Read Link
)

Problem 3 (25 points). The beginning of a free space bitmap looks like this after the disk partition is first formatted as :

1000 0000 0000 0000 0000

The first block is used by the root directory. The system always searches for free blocks **starting at the lowest numbered block** using **the contiguous allocation** and **the best-fit strategy**, so after writing file A, which uses 7 blocks, the bitmap looks like this:

1111 1111 1000 0000 0000.

Show the bitmap after each of the following additional actions (continue the table):

NN	ACTION	BITMAP																			
1	Initial state	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	File A is written, using 7 blocks	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
3	File B is written, using 5 blocks	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
4	File A is deleted	1	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
5	File C is written, using 7 blocks	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
6	File D is written, using 4 blocks	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
7	File C is deleted	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
8	File E is written, using 3 blocks	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
9	File F is written, using 1 block	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
10	File B is deleted	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0

(I think there must be a mistake in line 2, the should be 8 "1"s initially)

Problem 4 (20 points). What are the advantages and disadvantages of making the distributed systems? Divide into advantage, disadvantage and "not important" topics the following distributed system characteristics:

		Advantages	Disadvantages	Not important
1	Resource sharing	*		
2	Complexity		*	
3	Openness	*		
4	Concurrency	*		
5	Transparency	*		*
6	Security		*	
7	Fault tolerance	*		
8	Manageability		*	
9	Scalability	*		
10	Free-space Management			*
11	Recovery from failure	*		
12	Deadlock handling			*
13	Unpredictability		*	
14	Portability			*
15	Load balancing			* ?
16	Computation speedup	*		
17	Remote file transfer			*
18	Remote login			*
19	CPU scheduling			*
20	Thrashing			*
21	Failure Detection			*
22	Routing Strategies	*		
23	Paging			*
24	Demand Paging			*
25	Demand Segmentation			*

Remarks:

1. Mark the corresponding cell by "".*

2. You should compare standard (one CPU) and distributed systems. The term "Advantages" means that in distributed systems, the corresponding parameter is changed improving the quality and efficiency of operating systems services. The term "Not important" means that the corresponding parameter has no influence to distributed systems services.

3. To obtain 20 points it is enough to get 20 right answers.