CS349/611

Operating Systems

Spring 2016

**Written Assignment 3 (50 pts)**

**Due: March 31**

1. Assume that you have a page-reference string for a process with *m* frames (initially all empty). The page reference string has length *p*; n distinct page numbers occur in it. You can assume that p>n>m. Answer the questions (for any page replacement algorithms):
   1. What is the lower bound on the number of page faults? N all pages in reference string must page in
   2. What is an upper bound on the number of page faults? P every page in ref string causes a fault
2. Consider the following page-reference string:

1,2,3

,4,2,1,

5,|6,2,

1,2,3[],

7,6,3,

2,1,2,

3,6.

How many page faults would occur for the following replacement algorithms, assuming three, then four frames? Remember that all frames are initially empty, so your first unique pages will all cost one page fault. **Show your work**!

* LRU replacement
* FIFO replacement
* Optimal replacement

LRU 15 3 frames

1 1 1 4 4 5 5 5 1 1 7 7 2 2 2

2 2 2 2 2 6 6 6 3 3 3 3 3 3

3 3 1 1 1 2 2 2 2 6 6 1 6

LRU 10 4 frames

1 1 1 1 1 1 1 1 6 6

2 2 2 2 2 2 2 2 2

3 3 5 5 3 3 3 3

4 4 6 6 7 7 1

FIFO 16 3 frames

1 1 1 4 4 4 6 6 6 3 3 3 2 2 2 6

2 2 2 1 1 1 2 2 2 7 7 7 1 1 1

3 3 3 5 5 5 1 1 1 6 6 6 3 3

FIFO 14 4 frames

1 1 1 1 5 5 5 5 3 3 3 3 1 1

2 2 2 2 6 6 6 6 7 7 7 7 3

3 3 3 3 2 2 2 2 6 6 6 6

4 4 4 4 1 1 1 1 2 2 2

Optimal 11 3 Frames

1 1 1 1 1 1 3 3 3 3 3

2 2 2 2 2 2 7 2 2 2

3 4 5 6 6 6 6 1 6

Optimal 8 4 Frames

1 1 1 1 1 1 7 1

2 2 2 2 2 2 2

3 3 3 3 3 3

4 5 6 6 6

1. Consider the following two-dimensional array:

int A[][] = new int[100][100];

where A[0][0] is at location 200, in a paged system with pages of size 200. A small program code is in page 0 (locations from 0 to 199) for manipulating the matrix; thus, every instruction fetch will be from page 0.

For two page frames, how many page faults are generated by the following array-initialization loops, using a page replacement algorithm (the choice of replacement algorithm does not matter for this problem), and assuming that page frame 1 has the program code in it permanently, and the page frame 2 is initially empty (note also that arrays are stored on the hard drive row by row, and number of type int requires 2 bytes of memory – thus, one row of the array requires exactly one page of memory):

1. for(int j = 0; j<100; j++)

for (int i=0; i<100; i++)

A[i,j] = 0;

1. for (int i=; i<100; i++)

for (int j=0; j<100; j++)

A[i,j] =0;

. Explain your answer.

A: 10000 because it’s looping through the rows. For 2 frames it will be 100\*100 = 10000

B 100 because it’s read the whole row so only one page fault.