OS - Exercise #6

Lucas Hasting

November 2024

Contents

1 Question #7.2.1

A process in a paged system accesses the following virtual addresses: 10, 11, 104, 170, 73, 309, 185, 245, 246, 434, 458, 364.

Derive the corresponding reference string if the page size is 100 words, 200 words, and 300 words.

Solution: // will be used to refer to integer division.

100 words:

Reference String: 10 // 100, 11 // 100, 104 // 100, 170 // 100, 73 // 100, 309 // 100, 185 // 100, 245 // 100, 246 // 100, 434 // 100, 458 // 100, 364 // 100

Reference String: 0, 0, 1, 1, 0, 3, 1, 2, 2, 4, 4, 3

200 words:

Reference String: 10 // 200, 11 // 200, 104 // 200, 170 // 200, 73 // 200, 309 // 200, 185 // 200, 245 // 200, 246 // 200, 434 // 200, 458 // 200, 364 // 200

Reference String: 0, 0, 0, 0, 0, 1, 0, 1, 1, 2, 2, 1

300 words:

Reference String: 10 // 300, 11 // 300, 104 // 300, 170 // 300, 73 // 300, 309 // 300, 185 // 300, 245 // 300, 246 // 300, 434 // 300, 458 // 300, 364 // 300

Reference String: 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1

2 Question #7.2.3

A two-dimensional 512 x 512 array is stored in row major in a paged virtual memory with a page size of 512. Thus each row occupies 1 page. Physical memory contains fewer than 512 frames.

a) Using FIFO page replacement, determine how many page faults occur when accessing all elements sequentially:

1. in row major
2. in column major

b) Would the number of page faults be different under LRU page replacement?

Solution:

Assume the following for all general formulas: the matrix is n x n, n = page size, frame size < n.

a)

1. **Let x be the number of frames in physical memory**. Since $x < 512$, the number of page faults will depend on x. Since the items are being accessed sequentially, the general formula for page faults of row major order, accessed in sequential order is $page\ size - x$. **For this specific instance, the formula is 512 − x page faults**. An example with a smaller case is shown below:

   Let $x = 2$ (frame size) and let the array be a two-dimensional 4 x 4 array with a page size of 4.

   Using the formula, the expected value is $4 - 2 = 2$ page faults.

   Constructing the RS:

   Assuming the virtual addresses for the matrix are in sequential order starting at 0 and increasing for each column in the row. The RS for row major is shown below:

   row major RS: 0//4, 1//4, 2//4, 3//4, 4//4, 5//4, 6//4, 7//4, 8//4, 9//4, 10//4, 11//4, 12//4, 13//4, 14//4, 15//4

   row major RS: 0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3

   The table below traces through the FIFO algorithm, a "*" indicates a page fault.

   + is used to denote the page pointer.

Time	0	1	2	3	4	5	6	7	8	9*	10	11	12	13*	14	15	16
RS		0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
Frame 0	+0	+0	+0	+0	+0	+0	+0	+0	+0	2	2	2	2	+2	+2	+2	+2
Frame 1	1	1	1	1	1	1	1	1	1	+1	+1	+1	+1	3	3	3	3

   The above table shows 2 page faults which lines up with the expected value using the formula.

2. **Let x be the number of frames in physical memory**, since the columns are done in sequential order it will cycle between 0 through $page\ size - 1$ and will always have a page fault after all the frames have been used once, a general formula for page faults of column major order, accessed in sequential order is $(row\ size * col\ size) - x$. **For this specific instance, the formula is 262144 − x page faults**. ($512^2 = 262144$). An example of this formula is shown below:

   Let $x = 2$ (frame size) and let the array be a two-dimensional 4 x 4 array with a page size of 4.

   Using the formula, the expected value is $(4 * 4) - 2 = 16 - 2 = 14$ page faults.

   Constructing the RS:

   Assuming the virtual addresses for the matrix are in sequential order starting at 0 and increasing for each column in the row. The RS for row major is shown below:

column major RS: 0//4, 4//4, 8//4, 12//4, 1//4, 5//4, 9//4, 13//4, 2//4, 6//4, 10//4, 14//4, 3//4, 7//4, 11//4, 15//4

column major RS: 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3

The table below traces through the FIFO algorithm for the above RS, a "*" indicates a page fault.

+ is used to denote the page pointer.

Time	0	1	2	3*	4*	5*	6*	7*	8*	9*	10*	11*	12*	13*	14*	15*	16*
RS		0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
Frame 0	+0	+0	+0	2	+2	0	+0	2	+2	0	+0	2	+2	0	+0	2	+2
Frame 1	1	1	1	+1	3	+3	1	+1	3	+3	1	+1	3	+3	1	+1	3

The above table shows 14 page faults which lines up with the expected value using the formula.

b) **Both row major and column major end up with the same formulas as part "a" using LRU page replacement, and therefore has the same number of page faults as FIFO**. The reason this is the case is because the amount of swapping, regardless of either algorithm (with sequential ordering of rows or columns), the amount of swapping is the same (depending on row major or column major order). Examples of this are shown below, they are the same examples as shown in "a", but they use the LRU page replacement algorithm instead of the FIFO algorithm.

Example 1: row major

Recall the number of page faults from example 1 in part "a" was 2 page faults, the table below traces through the LRU page replacement algorithm for the first example in part "a", a "*" indicates a page fault.

Time	0	1	2	3	4	5	6	7	8	9*	10	11	12	13*	14	15	16
RS		0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
Frame 0	0	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2
Frame 1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	3	3	3
Queue Head	0	1	1	1	1	0	0	0	0	1	1	1	1	2	2	2	2
Queue End	1	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3

The above table shows 2 page faults which lines up with the first example in part "a".

Example 2: column major

Recall the number of page faults from example 2 in part "a" was 14 page faults, the table below traces through the LRU page replacement algorithm for the second example in part "a", a "*" indicates a page fault.

Time	0	1	2	3*	4*	5*	6*	7*	8*	9*	10*	11*	12*	13*	14*	15*	16*
RS		0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
Frame 0	0	0	0	2	2	0	0	2	2	0	0	2	2	0	0	2	2
Frame 1	1	1	1	1	3	3	1	1	3	3	1	1	3	3	1	1	3
Queue Head	0	1	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2
Queue End	1	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3

The above table shows 14 page faults which lines up with the second example in part "a".

3 Question #7.3.2

Physical memory consists of 4 page frames, initially all empty. The following reference string is processed:
0 1 4 0 2 3 0 1 0 2 3 4 2 3

a) Show which pages are resident under the second chance page replacement algorithm. Indicate when page faults occur.

b) Assume that references to page 1 are write references (modifying page 1) and all others are read references. Show which pages are resident under the third chance page replacement algorithm. Indicate when page faults occur.

Solution:

a) The algorithm is shown using the tables below, page faults occur when a "*" is present, and + is used to denote the page pointer:

Time	0		1*		2*		3*		4		5*		6*		7*	
RS			0		1		4		0		2		3		0	
Frame 0	+		0	1	0	1	0	1	0	1	+0	1	3	1	3	1
Frame 1			+		1	1	1	1	1	1	1	1	+1	0	0	1
Frame 2					+		4	1	4	1	4	1	4	0	+4	0
Frame 3							+		+		2	1	2	0	2	0

Time	8*		9		10		11		12*		13*		14*	
RS	1		0		2		3		4		2		3	
Frame 0	3	1	3	1	3	1	3	1	+3	0	2	1	2	1
Frame 1	0	1	0	1	0	1	0	1	0	0	+0	0	3	1
Frame 2	1	1	1	1	1	1	1	1	1	0	1	0	+1	0
Frame 3	+2	0	+2	0	+2	1	+2	1	4	1	4	1	4	1

b) The algorithm is shown using the table below, page faults occur when a "*" is present, and + is used to denote the page pointer:

Time	0		1*		2*		3*		4		5*		6*		7*	
RS			0		1w		4		0		2		3		0	
Frame 0	+		0	10	0	10	0	10	0	10	+0	10	3	10	3	10
Frame 1			+		1	11	1	11	1	11	1	11	+1	01	1	00
Frame 2					+		4	10	4	10	4	10	4	00	0	10
Frame 3							+		+		2	10	2	00	+2	00

Time	8		9		10		11		12*		13*		14*	
RS	1w		0		2		3		4		2		3	
Frame 0	3	10	3	10	3	10	3	10	+3	00	2	10	2	10
Frame 1	1	11	1	11	1	11	1	11	1	01	+1	01	1	00
Frame 2	0	10	0	10	0	10	0	10	0	00	0	00	3	10
Frame 3	+2	00	+2	00	+2	10	+2	10	4	10	4	10	+4	10

4 Question #7.4.1

Physical memory is initially empty. The following reference string is processed: 0 1 4 0 2 3 0 1 0 2 3 4 2 3

a) Show which pages are resident in an optimal working set with d = 3. Indicate when page faults occur. Determine the average working set size.

b) Show which pages are resident under the working set page replacement algorithm with d = 3. Indicate when page faults occur. Determine the average working set size.

Solution:

a) The optimal working set algorithm is traced using the table below, a "*" indicates a page fault.

Time	0	1*	2	3*	4*	5	6*	7	8*	9*	10*	11	12	13	14
RS		0	1	4	0	2	3	0	1	0	2	3	4	2	3
Optimal Working Set		0	1	4	0	2	3	0	1	0	2	3	4		
		1	4	0	2	3	0	1	0	2	3	4	2		
		4	0	2	3	0	1		2	3	4	2	3		
Size		3	3	3	3	3	3	2	3	3	3	3	3		

The average working set size (A) is shown below

$$A = \frac{11 * 3 + 2}{12} = \frac{33 + 2}{12} = \frac{35}{12} \approx 2.917$$

b) The working set page replacement algorithm is traced using the table below, a "*" indicates a page fault.

Time	-1*	0*	1*	2	3*	4*	5	6*	7	8*	9*	10*	11	12
RS	0	1	4	0	2	3	0	1	0	2	3	4	2	3
Page 0	x	x	x	x	x	x	x	x	x	x	x	-	-	-
Page 1	-	x	x	x	-	-	-	x	x	x	-	-	-	-
Page 2	-	-	-	-	x	x	x	-	-	x	x	x	x	x
Page 3	-	-	-	-	-	x	x	x	-	-	x	x	x	x
Page 4	-	-	x	x	x	-	-	-	-	-	-	x	x	x
Size	1	2	3	3	3	3	3	3	2	3	3	3	3	3

The average working set size (A) is shown below

$$A = \frac{11 * 3 + 2 * 2 + 1}{14} = \frac{33 + 4 + 1}{14} = \frac{38}{14} \approx 2.714$$

5 Question #7.4.3

For d = 5 and the reference string 1 0 0 0 0 0 0 1 1 3 0 0 0 1 5 1 5 4 4 2 0 0 4 4.

a) What is the largest working set the reference string will require?

b) What is the smallest working set the reference string will require?

Solution:

Both a and b are solved by tracing the optimal working set algorithm using the tables below:

Time	0	1	2	3	4	5	6	7	8	9	10	11	12
RS		1	0	0	0	0	0	0	1	1	3	0	0
Optimal Working Set		1 0	0	0	0 1	0 1	0 1 3	1 3 0	1 3 0	1 3 0	3 0 1	0 1 5	0 5 1
Size		2	1	1	2	2	3	3	3	3	3	3	3

Time	13	14	15	16	17	18	19	20	21	22	23	24
RS	0	1	5	1	5	4	4	2	0	0	4	4
Optimal Working Set	0 1 5	1 5 4	1 5 4	1 5 4 2	5 4 2 0	4 2 0	2 0 4	2 0 4				
Size	3	3	3	4	4	3	4	4				

Therefore, the largest working set is 4, and the smallest working set is 1.