COS214 Tutorial 5

Roberto Togneri, 2000

1. Discuss two approaches to the replacement of pages in a paged memory system. What information is required to optimise the decisions made in the algorithms you describe? Consider the following sequence of page references: 1 2 3 4 4 2 1 4 1 3 4. Determine how many page faults will occur for each algorithm, assuming there are only 2 physical page frames and that both are initially invalid.

2. A segmented memory has paged segments, with each virtual address having a 2-bit segment number, a 2-bit page number and an 11-bit address within the page. The main memory contains 32Kb. Each segment is either read-only (ro), read execute (rx), read-write (rw) or read-write-execute (rwx). The page table and protection is as follows:

Seg 0		**Seg 1**		**Seg 2**		**Seg 3**	
ro		**rx**		**rwx**		**rw**	
Virtual Page	Page Frame	Virtual Page	Page Frame	Virtual Page	Page Frame	Virtual Page	Page Frame
0	9	0	disk	Page table not in memory		0	14
1	3	1	0			1	1
2	disk	2	15			2	6
3	12	3	8			3	disk

   (a) How many virtual pages are there in the address space?
   (b) How many physical pages can fit in main memory?
   (c) For each of the following indicate what physical address is computed from the virtual address, if any. If a segment, page or protection fault occurs specify which one.

No	**Access**	**Seg**	**Page**	**Offset**
1	read	0	1	1
2	read	1	1	10
3	read	3	3	2047
4	write	0	1	4
5	write	2	1	2
6	write	1	0	14
7	jump to	1	3	100
8	read	0	2	50
9	read	2	0	5
10	jump to	3	0	60

3. Page replacement algorithms work best with hardware support. For example, many systems provide *Referenced (R)* and *Modified (M)* flags in hardware. Explain how these flags give the name *Second-Chance FIFO*. Next, explain how the flags could be simulated at only a small amount of expense in software.

4. A computer provides each process with 64K of address space divided into pages of 4K. A particular program has a text size of 32,768 bytes, a data size of 16,386 bytes, and a stack size of 15,870 bytes. Will this program fit in the address space? If the page size were 512 bytes, would it fit? Explain! Remember that a page may not contain parts of two different segments.

5. Which of the following programming techniques and structures are “good” for a demand-paged environment ? Which are “not good”? Explain your answers.
   (a) Stack
   (b) Hashed symbol table
   (c) Sequential search
   (d) Binary search
   (e) Pure code
   (f) Vector operations
   (g) Indirection

6. An operating system supports a paged virtual memory, using a central processor with a cycle time of 1 microsecond. It costs an additional 1 microsecond to access a page other than the current one. Pages have 1000 words, and the paging device is a drum that rotates at 3000 revolutions per minute, and transfers 1 million words per second. The following statistical measurements were obtained from the system:
   - 1 percent of all instructions executed accessed a page other than the current page.
   - Of the instructions that accessed another page, 80 percent accessed a page already in memory.
   - When a new page was required, the replaced page was modified 50 percent of the time.

   Calculate the effective instruction time on this system, assuming that the system is running one process only, and that the processor is idle during drum transfers. Assume that the drum rotational delay is half a revolution and seek time is negligible.

7. Consider the following page table:

Index	f	P	time of last access
0	--	0	--
1	0	1	100
2	--	0	--
3	2	1	99
4	--	0	--
5	--	0	--
6	1	1	80
7	3	1	115

   What is the PA if VA = [p = 2 | d = 233] is referenced at time 120 and pages are 1K? Assume local scope LRU page replacement. Describe any changes made to the page table.

8. A 64-bit CPU has a VA address range of 2^{64} = 4GB x 4GB which is some really, really, really huge number. What minimum level of page tables do you think is reasonable assuming a 16K page? Comment on the performance cost of adopting such a scheme and is this an issue?