Exercise 3 – Memory magagement. Virtual memory

Questions are taken from Stallings, Operating Systems Internals and Design Principles, fifth edition and Silberschatz et al., Operating System Concepts, seventh edition.

1 – Stallings 7.15

Consider a memory in which contiguous segments $S_1, S_2, ...S_n$ are placed in their order of creation from one end of the store to the other, as suggested by the following figure:

When segment S_{n+1} is being created, it is placed immediately after segment S_n even though some of the segments $S_1, S_2, ...S_n$ may already have been deleted. When the boundary between segments (in use or deleted) and the hole reaches the other end of the memory, the segments in use are compacted.

a) Show that the fraction of time F spent on compacting obeys the following inequality:

$$F \geq \frac{1-f}{1+kf} \text{ where } k = \frac{t}{2s} - 1 \tag{1}$$

s = average length of a segment, in words

t = average lifetime of a segment, in memory references

f = fraction of the memory that is unused under equilibrium conditions

Hint: Find the average speed at which the boundary crosses the memory and assume that the copying of a single word requires at least two memory references.

b) Find F for $f = 0.2$, $t = 1000$, and $s = 50$.

2 – Stallings 8.10

Assuming a page size of 4 Kbytes and that a page table entry takes 4 bytes, how many levels of page tables would be required to map a 64-bit address space if the top-level page table fits into a single page?

3 – Stallings 8.19

The UNIX kernel will dynamically grow a process's stack in virtual memory as needed, but it will never try to shrink it. Consider the case in which a program calls a C subroutine that allocates a local array on the stack that consumes 10K. The kernel will expand the stack segment to accommodate it. When the subroutine returns, the stack pointer is adjusted and this space could be released by the kernel, but it is not released. Explain why it would be possible to shrink the stack at this point and why the UNIX kernel does not shrink it.

4 –Silberschatz 9.5

The following data is given for a system:
Demand paging, with the page table in internal registers. The memory access time (T_{ma}) is 100 ns. A page fault and the following read operation requires 8 ms plus another 12 ms if a modified page is replaced. Assume that 70% of the replaced pages is modified.
What is the maximum page fault frequency that can be accepted if an effective access time of maximum 200 ns is wanted?

5

Compare the number of page faults with the replacement algorithms LRU, FIFO and OPT (Optimal), if the following sequence of page references (reference string) is used:
1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6
Investigate some of the cases with 1,2,3,4,5,6 or 7 page frames available.

6 –Silberschatz 9.10

A measurement on a demand paging system has given the following utilizations:

- CPU utilization: 20%
- Paging disk: 97.7%
- Other I/O devices: 5%

For each of the following, explain whether it is likely to improve the CPU utilization:

a) Install a faster CPU.

b) Install a bigger paging disk.

c) Increase the degree of multiprogramming.

d) Decrease the degree of multiprogramming.

e) Install more main memory.

f) Install a faster hard disk or multiple controllers with multiple hard disks.

g) Add prepaging to the page-fetch algorithm.

h) Increase the page size.