Homework4

马俊杰PB20051093

1.Explain the following terms：

Segmentation Fault:　即段错误，当进行非法地址访问时OS返回的错误类型

TLB:　即Translation lookaside buffer，是关联的高速缓存，用于存储少量页表条目，当CPU产生一个逻辑地址后先在TLB中寻找；这样的用于快速查找帧码的高速硬件缓冲就是TLB

Page Fault:　在页式地址映射过程中，根据要访问的逻辑地址无法映射到可访问的物理页时发生的异常。即当进程试图访问那些尚未调入内存的页面时产生的错误

Demand paging: 当某页的内容被需要的时候才从外存调入内存的一种存储管理技术。

2.Introduce the concept of thrashing, and explain under what circumstance thrashing will happen.

在虚拟存储器中，刚从主存中换出的页面序列，很快又被调入的页面序列反复调进调出的现象。这种高度的页面调度活动称为抖动

当CPU调度程序因为CPU利用率降低而增加多道程度，使得分配给进程的帧小于进程所需的帧数，导致更多的缺页错误和更长的调页设备队列，进而试图再次增加多道程度，使得缺页错误率显著增加，系统始终忙于调页，即系统抖动

3.Consider a paging system with the page table stored in memory.

a. If a memory reference takes 50 nanoseconds, how long does a paged memory reference take?

首先需要访问位于内存中的页表，得到帧码后再访问内存，因此需要100ns

b. If we add TLBs, and 75 percent of all page-table references are found in the TLBs, what is the effective memory reference time? (Assume that finding a page-table entry in the TLBs takes 2 nanoseconds, if the entry is present.)

首先得到帧码，无论是否命中，都需要2ns，然后25%需要在内存中访问页表，最后访问内存

$$\text{有效访问时间} = 2ns + 25\% \times 50ns + 50ns = 64.5ns$$

4.Assume a program has just referenced an address in virtual memory. Describe a scenario how each of the following can occur: (If a scenario cannot occur, explain why.)

- TLB miss with no page fault

  TLB中无对应页码，但页表中存在对应页码，得到帧码计算出物理地址，无缺页错误

- TLB miss and page fault

TLB中无对应页码，同时在页表中也未找到对应页码，引发缺页错误

- TLB hit and no page fault

  TLB中有对应页码，无缺页错误

- TLB hit and page fault

  不会发生，TLB中有页码代表已经加载入内存，不会发生缺页错误

5.Assume we have a demand-paged memory. The page table is held in registers. It takes 8 milliseconds to service a page fault if an empty page is available or the replaced page is not modified, and 20 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Assume that the page to be replaced is modified 70 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200 nanoseconds?

假设缺页错误率为p

$$\text{有效访问时间} = (1-p) \times 100ns + p \times (0.3 \times 8ms + 0.7 \times 20ms) \leq 200ns$$

$$\therefore p \leq \frac{1}{163} \approx 0.61\%$$

6.Consider the following page reference string: 7, 2, 3, 1, 2, 5, 3, 4, 6, 7, 7, 1, 0, 5, 4, 6, 2, 3, 0, 1. Assuming demand paging with three frames, how many page faults would occur for the following replacement algorithms?

- LRU replacement

7	7	7	1		1	3	3	3	7		7	7	5	5	5	2	2	2	1
	2	2	2		2	2	4	4	4		1	1	1	4	4	4	3	3	3
		3	3		5	5	5	6	6		6	0	0	0	6	6	6	0	0

  因此出现18次缺页错误

- FIFO replacement

7	7	7	1		1		1	6	6		6	0	0	0	6	6	6	0	0
	2	2	2		5		5	5	7		7	7	5	5	5	2	2	2	1
		3	3		3		4	4	4		1	1	1	4	4	4	3	3	3

  因此出现17次缺页错误

- Optimal replacement

7	7	7	1		1		1	1	1			1		1	1	1	1		
	2	2	2		5		5	5	5			5		4	6	2	3		
		3	3		3		4	6	7			0		0	0	0	0		

因此出现13次缺页错误

7.Explain what Belady’s anomaly is, and what is the feature of stack algorithms which never exhibit Belady’s anomaly?

Belady异常：采用FIFO算法时，如果对一个进程未分配它所要求的全部页面，有时就会出现分配的帧数增多但缺页率反而提高的异常现象。

即当$n<m$时，采用堆栈算法得到的n页框的集合$\subset$m页框的集合，此时若m页框出现缺页错误，n页框一定也出现，因此不会出现Belady异常