Page Replacement Algorithms

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

1. TLB miss with no page fault
2. TLB miss and page fault
3. TLB hit and no page fault
4. TLB hit and page fault

Answer:
1. TLB miss with no page fault page has been brought into memory but has been removed from the TLB.
2. TLB miss and page fault has occurred.
3. TLB hit and no page fault page is in memory and in the TLB. Most likely a recent reference.
4. TLB hit and page fault cannot occur. The TLB is a cache of the page table. If an entry is not in the page table, it will not be in the TLB.

Example
Consider the following page reference string:

7, 2, 3, 1, 2, 5, 3, 4, 6, 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
- FIFO replacement
- Optimal replacement

Answer:
- 18
- 17
- 13

Exercise
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 frames? Remember all frames are initially empty, so your first unique pages will all cost one fault each.

- LRU replacement
- FIFO replacement
- Optimal replacement
Clock Algorithm

In Second-chance (clock) page-replacement algorithm the reference bit for a page is set by the hardware whenever that page is referenced (either a read or a write to any byte in the page).

Initially, all bits are cleared (to 0) by the operating system. As a user process executes, the bit associated with each page referenced is set (to 1) by the hardware. After some time, we can determine which pages have been used and which have not been used by examining the reference bits.

The basic algorithm of second-chance replacement is a FIFO replacement algorithm. When a page has been selected, however, we inspect its reference bit. If the value is 0, we proceed to replace this page; but if the reference bit is set to 1, we give the page a second chance and move on to select the next FIFO page. When a page gets a second chance, its reference bit is cleared, and its arrival time is reset to the current time. Thus, a page that is given a second chance will not be replaced until all other pages have been replaced (or given second chances).

One way to implement the second-chance algorithm (sometimes referred to as the clock algorithm) is as a circular queue. A pointer (that is, a hand on the clock) indicates which page is to be replaced next. When a frame is needed, the pointer advances until it finds a page with a 0 reference bit. As it advances, it clears the reference bits. Once a victim page is found, the page is replaced, and the new page is inserted in the circular queue in that position.

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

Exercise:
16. Suppose we have the following page accesses: 1 2 3 4 2 3 4 1 2 1 3 1 4 and that there are three frames within our system. Using the FIFO replacement algorithm, what is the number of page faults for the given reference string?
   A) 14  
   B) 8  
   C) 13  
   D) 10

17. Suppose we have the following page accesses: 1 2 3 4 2 3 4 1 2 1 3 1 4 and that there are three frames within our system. Using the FIFO replacement algorithm, what will be the final configuration of the three frames following the execution of the given reference string?
   A) 4, 1, 3  
   B) 3, 1, 4  
   C) 4, 2, 3  
   D) 3, 4, 2
18. Suppose we have the following page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and that there are three frames within our system. Using the LRU replacement algorithm, what is the number of page faults for the given reference string?

A) 14  
B) 13  
C) 8  
D) 10

19. Given the reference string of page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and a system with three page frames, what is the final configuration of the three frames after the LRU algorithm is applied?

A) 1, 3, 4  
B) 3, 1, 4  
C) 4, 1, 2  
D) 1, 2, 3

20. Optimal page replacement ____.

A) is the page-replacement algorithm most often implemented  
B) is used mostly for comparison with other page-replacement schemes  
C) can suffer from Belady's anomaly  
D) requires that the system keep track of previously used pages

21. In the enhanced second chance algorithm, which of the following ordered pairs represents a page that would be the best choice for replacement?

A) (0,0)  
B) (0,1)  
C) (1,0)  
D) (1,1)

22. _____ is the algorithm implemented on most systems.

A) FIFO  
B) Least frequently used  
C) Most frequently used  
D) LRU

**Trashing**

What is the cause of thrashing? How does the system detect thrashing? Once it detects thrashing, what can the system do to eliminate this problem?

**Answer:** Thrashing is caused by underallocation of the minimum number of pages required by a process, forcing it to continuously page fault. The system can detect thrashing by evaluating the level of CPU utilization as compared to the level of multiprogramming. It can be eliminated by reducing the level of multiprogramming.
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>B</td>
</tr>
<tr>
<td>17</td>
<td>D</td>
</tr>
<tr>
<td>18</td>
<td>C</td>
</tr>
<tr>
<td>19</td>
<td>B</td>
</tr>
<tr>
<td>20</td>
<td>B</td>
</tr>
<tr>
<td>21</td>
<td>A</td>
</tr>
<tr>
<td>22</td>
<td>D</td>
</tr>
</tbody>
</table>