**T1 2024: ICT201 Computer Organisation and Architecture**

**Tutorial 08**

**Topic: Week 8 Virtual Memory**

**Submission: Five minutes before the end of the tutorial: all files must be uploaded on Moodle.**

**Exercise 1:**

Estimated completion time: **45 minutes**

1. What is demand paging?
2. How does the use of virtual memory improve system utilization?
3. Why is the principle of locality crucial to the use of virtual memory?
4. Which considerations determine the size of a page?
5. What is the purpose of a translation lookaside buffer?
6. What are the drawbacks of using either only a precleaning policy or only a demand cleaning policy?
7. Explain thrashing.
8. What is the relationship between FIFO and clock page replacement algorithms?
9. How is a page fault trap dealt with?
10. Why is it not possible to combine a global replacement policy and a fixed allocation policy?

**Exercise 2 (Activity Task):**

Estimated completion time: **25 minutes**

Students should answer this exercise activity during the tutorial session. Students should support their answer and explain their understanding.

*It is important that students be active in the class. Students who is not active will be consider as absent even if already logged in to the session.*

|  |  |  |
| --- | --- | --- |
| T | F | 1) The size of virtual storage is limited by the actual number of main storage locations. |
|  |  |  |
| T | F | 2) The addresses a program may use to reference memory are distinguished from the addresses the memory system uses to identify physical storage sites. |
|  |  |  |
| T | F | 3) Most of the memory management issues confronting the operating system designer are in the area of paging when segmentation is combined with paging. |
|  |  |  |
| T | F | 4) Segmentation is not visible to the programmer. |
|  |  |  |
| T | F | 5) The placement policy determines where in real memory a process piece is to reside. |
|  |  |  |
| T | F | 6) Virtual memory allows for very effective multiprogramming and relieves the user of the unnecessarily tight constraints of main memory. |
|  |  |  |
| T | F | 7) The principle of locality states that program and data references within a process do not tend to cluster. |
|  |  |  |
| T | F | 8) The smaller the page size, the greater the amount of internal fragmentation. |
|  |  |  |
| T | F | 9) The design issue of page size is related to the size of physical main memory and program size. |
|  |  |  |
| T | F | 10) Segments may be of unequal, indeed dynamic, size. |
|  |  |  |
| T | F | 11) The page currently stored in a frame may still be replaced even when the page is locked. |
|  |  |  |
| T | F | 12) One way to counter the potential performance problems of a variable-allocation global scope policy is to use page buffering. |
|  |  |  |
| T | F | 13) The PFF policy evaluates the working set of a process at sampling instances based on elapsed virtual time. |
|  |  |  |
| T | F | 14) A precleaning policy writes modified pages before their page frames are needed so that pages can be written out in batches. |

**Exercise 3:**

Estimated completion time: **40 minutes**

* Watch the YouTube videos provided on the Moodle under week 8 and discuss in group.