University Mohamed Khider Biskra **Faculty of exact sciences and natural and life sciences.** **Department of Computer Science**		
Level : 2LMD	**Date: March 2023**	**Module** : Operating Systems 1
Series 2: Non-contiguous memory management		

Exercise 1

Consider a program whose code occupies 1024 bytes in memory and which uses a vector with 1000 elements of character type (one character = 2 bytes). This program is executed in a system that uses memory paging whose actual memory size is 1 MB, the size of a page is 512 bytes, and the memory-referenced instructions have an address field of 24 bits.

I) Give:

1) the size of the logical address space
2) the number of bits of the offset
3) the number of bits of the virtual page number
4) the number of bits of a physical address
5) the number of bits of the real page number (frame)
6) the number of entries in the page table

II) Does the loading of this program into memory cause an internal fragmentation? Justify your answer.

III) Give the format of a 32-bit virtual address with 256-byte pages and a three-level page table. If all tables have the same number of entries, give the number of tables as well as their number of entries.

Exercise 2

Consider the sequence of memory references for a program of size 460 words:

10 11 104 170 73 309 185 245 246 434 458 364

Assuming that the size of a page is 100 words and the size of the available main memory (MM) is 200 words. In the case of FIFO, LRU, and optimal replacement algorithms give:

1. The number of page faults.
2. The page fault rate.
3. The number of wasted words.

Exercise 3

We propose to write the algorithm of replacement of the pages according to their chronological order of use (LRU algorithm). To do this, we have a sequence of memory references, a program of given size T and a main memory MM divided into pages each having a size Spage.

1. Choose the data structures necessary to implement the algorithm.
2. Write the LRU algorithm.
3. Discuss this solution with other LRU implementation algorithms

Exercise 4

Suppose a segmented and paged memory. Each virtual address has a 2-bit segment number, a 2-bit page number and an 11-bit page offset. The main memory contains 32 KB, divided into 2 KB pages. Each segment is either read-only, read-write or read-write-execute. The following page and protection table is available:

Segment0		**Segment1**		**Segment2**		**Segment3**	
Read-only		Read-execution		Read.-write.-execution		Read-write	
Virtual Page	Frame	Virtual Page	frame	Virtual Page	frame	Virtual Page	frame
0	9	0	on disk	The page table is not in main memory		0	14
1	3	1	0			1	1
2	On disk	2	15			2	6
3	12	3	8			3	On disk

For each of the following accesses to the virtual memory, give the calculated physical address. If an error occurs, give its type.

Access	**Segment**	**Page**	**Offset**
1. Load data	0	1	1
2. Load data	1	1	10
3. Load data	3	3	2047
4. Save data	0	1	4
5. Data storage	3	1	2
6. Save data	3	0	14
7. Jump to the address	1	3	100
8. Load data	0	2	50
9. Read data	2	0	5
10. Jump to the address	3	0	60

Exercise 5

We consider a system with a segmented logical memory with a page size of 4 kB and a physical memory of 64 kB. The address space of a process P is composed of three segments S1, S2 and S3 of size, respectively 16 KB, 8 KB and 4 KB. At a given time, for process P, pages 2 and 3 of segment S1, page 2 of segment S2 and page 1 of segment S3 are loaded in physical memory, respectively in frames 2, 0, 9, 12. For a data located in the address space of process P at decimal address 8212.

a) Describe the principle of paging applied to memory segmentation. How is the translation of a virtual address performed?

b) Indicate:

1. the segment
2. the page number in the segment
3. the offset in the page
4. the frame number
5. the offset in the frame
6. the physical address (in decimal and in binary)