SURNAME		NAME		Group
ID		Signatura		

- **Keep the exam sheets stapled.**
- **Write your answer inside the reserved space.**
- **Use clear and understandable writing. Answer briefly and precisely.**
- **The exam has 9 questions, everyone has its score specified.**

---

1. A paging system has pages of 4KB, 28-bit logical addresses and 4GB of main memory. In this system a process is being executed and the following table shows the **partial** content (in hexadecimal) of the process page table:

Page	Frame	Bv
0	521F5	1
1	112D2	1
2	43231	1
3		0
4	231F2	1
5	231F3	1
6	231F4	1
...	...	...

(1,0 points = 0,5 + 0,5)

1	**a)** Compute, in hexadecimal, the physical address generated if the process emits the logical address 0002F1F (in hexadecimal). Explain concisely your answer.
	b) Compute, in hexadecimal, the logical address generated if the process does access to the physical address 231F2288 (in hexadecimal). Explain concisely your answer.

2. An operating system has to manage efficiently the computer physical memory. Think about the problems that memory management has and answer the following items:

(1,0 points = 0,5 + 0,5)

2	**a)** What problem intends to solve the technique of compacting memory and on what type or types of contiguous memory allocation schemes can be applied?
	b) What is the utility of the MMU (Memory Unit Manager) and what problems has it to solve?

3. Considering the memory map of a UNIX process, indicate whether the following statements are true(T) or false(F):

(0,6 points)

3	STATEMENT	T/F
	The uninitialized data region generated when running a program, is a region without support in any file.	
	The code region with support on the executable file of a process, has read, write, and execute permissions.	
	When a process maps a file into memory a new region is created in the memory map of this process that is accessed using input/output calls. .	
	The use of dynamic libraries is based on the technique of mapping libraries in memory during execution of the process that uses them.	

4. Consider an operating system that manages paged virtual memory with 4 KB pages. Free frames are assigned in ascending order of physical addresses using local replacement policy. The system assigns three frames to a new process (frames 0 to 2) and during the process execution the following logical addresses are issued (in hexadecimal):

400000, 400014, 800000, 800024, 600030, 600034, 800280, 40060c, 200000, 400c10, 600f24

Indicate which pages stay in physical memory after the last access and what is the last physical address that is accessed in the following cases. (Indicate with '—' both the free frames or the busy frames with an invalid page).

(2,0 points = 0,5 + 0,5 + 0,5 + 0,5)

4

a) LRU replacement algorithm

Frame	Page
0	
1	
2	

Last physical address:

b) Second chance replacement algorithm

Frame	Page
0	
1	
2	

Last physical address:

c) Optimal replacement algorithm

Frame	Page
0	
1	
2	

Last physical address:

d) Suppose now that a FIFO algorithm is applied and after the last access on the sequence, the process does an exec call and after that the following logical addresses are emitted:
400000, 400010, 600020

Frame	Page
0	
1	
2	

Last physical address:

5. In a two level paging system with 32-bit physical and logical addresses, page size is 4KB and the second level page tables are allocated in main memory occupying one frame each one.

(1,0 points = 0,5 + 0,5)

5	**a)** Determine the number of entries that will have every second level page table. Consider that each table entry or page descriptor includes 12 control bits like Bv, Br, Bm, R, W, X, etc.
	b) Describe the structure of the logical addresses and the physical addresses in this system, and the size in bits of every address field .

6. The content of the file descriptor tables corresponding to three processes at a given time are the following:

Initial page table	
0	STDIN
1	STDOUT
2	STDERR
3	
4	
5	

P1 page table	
0	STDIN
1	STDOUT
2	STDERR
3	"result"
4	"fd_pipe[0]"
5	"fd_pipe[1]"

P2 page table	
0	STDIN
1	"fd_pipe[1]"
2	STDERR
3	
4	
5	

P3 page table	
0	"fd_pipe[0]"
1	"result"
2	STDERR
3	
4	
5	

Complete the following C code fragment with POSIX primitives required to create the three processes and to match the content shown of their descriptor tables, at points marked as /*** Table ...***/

(1.0 points)

6

```
int fd_pipe[2]; /* pipe descriptor */
int fd;     /* regular file descriptor */

/*** Initial descriptor table ***/
  fd=open("result",O_WRONLY |O_CREAT|O_TRUNC,0666);


 /*** P1 descriptor table ***/
  if (!(pid=fork())) {









 /*** P2 descriptor table ***/
    execlp("/bin/cat", "cat", "fich1", NULL);
  }
  if(!(pid=fork())){









  /*** P3 descriptor table ***/
    execlp("/usr/bin/wc", "wc", "-l",NULL);
  }
  close(fd_pipe[0]); close(fd_pipe[1]);
  close (fd);
  while(pid != wait(&status));
}
```

7. Given the following directory listing on a POSIX system:

```
i-node permissions  links   user      group size       date         name
2021662 drwxr-xr-x   2      sterrasa  fso  4096        dec  9 2015  .
2021611 drwxr-xr-x   8      sterrasa  fso  4096        sep 10 2015  ..
2021663 -rwxr-sr-x   1      sterrasa  fso  1139706     dec  9 2015  copia
2021664 -rw-rw-r--   1      sterrasa  fso  9706        dec  9 2015  f1
2021665 -rw-rw-r--   2      sterrasa  fso  4157        dec  9 2015  f2
2021665 -r--r--rw-   2      sterrasa  fso  4157        dec  9 2015  f3
2021666 lrwxrwxrwx   1      sterrasa  fso     2        dec  9 2015  f4->f1
```

(1.25 points = 0.75 + 0.5)

7

a) The program copy copies the file contents received as the first argument into another file whose name is indicated as the second argument. That is, "copy a b" copies the contents of file *a* to file *b*, if b does not exist it has to create it and then perform the copy. Fill the following table, indicating in case of success what permissions are checked and, in case of failure, which is the permission missed and why.

(UID,GUI)	COMMAND	DOES IT WORK?	EXPLANATION
(pepe, fso)	copia f1 f5		
(sterrasa, fso)	copia f1 f3		
(ana, etc)	copia f3 f4		

b) Indicate what values from column `links` on the former listing will change if directory entry *f3* is removed (`rm f3`). What values of the same column will change if removing *f4*?. Explain your answers.

8. A 32MByte partition has 2MBytes reserved to file system control structures and it is organized into 512-byte blocks, the size of a pointer to block is. Obtain:

(1,0 points = 0,5 + 0,5)

8	**a)** Maximum file size using indexed allocation with a one block of pointers per file.
	b) Maximum file size using linked allocation.

9. A disk with 6GB capacity, is formatted with a version of MINIX whose sizes are as follows :

- The boot block and superblock occupy one block each.
- The i-node size is 32 bytes (7 direct pointers, 1 indirect, 1 double indirect).
- With 16-bit zone pointers.
- Every directory entry is 16 bytes.
- 1 zone = 1 block = 1 KByte
- The number of reserved i-nodes is 4.096

The scheme of the partition elements is the following:

Boot	Superblock	i-node bi map	Zone bitmap	i-nodes	Data area

Obtain:

a) Number of blocks occupied by the following elements: the i-node bitmap, the zone bitmap and the inodes.

b) The content of the existing directories is shown, assuming that every regular files occupies 10 KBytes, how many inodes are busy? how many data zones are occupied?

1	.
1	..
4	usr
10	bin

4	.
1	..
20	list
12	alumno

12	.
4	..
26	prac

10	.
1	..
35	calc

(1,15 points = 0,5 + 0,65)

9	a)
	b)