

Juan Valentín Guerrero Cano 45338112Y

$$h=8 \quad L=1 \quad M=1 \quad N=2$$

### Tema 3

① a) dirección de 10 / 10 / 12 bits.

Código: INICIO: 4096 ~~2~~ FIN: 12287  
8192

DATOS: INICIO: 12288 FIN: 18431

PILA HEBRA: BASE:  $2^{30} - 2^{13} + 12 \cdot 2^{10} - 1$  Tam: 2048+

PILA MAIN: B:  $2^{30} - 1$  Tam:  $1204 \cdot 8 + 4096$

PILA HEBRA: INICIO:  $2^{30} - 2^{13} + 12 \cdot 2^{10} - 1 - 2048 - 1024$

FIN:  $2^{30} - 2^{13} + 12 \cdot 2^{10} - 1$

PILA MAIN: INICIO:  $2^{30} - 1 - 1204 \cdot 8 - 4096$

FIN:  $2^{30} - 1$

b) ~~TP 1°N  $\rightarrow 2^{10}$  entradas (1024)~~  
~~TP 2°N  $\rightarrow 2^{10}$  entradas (1024)~~

Joa Valentin Guerrero Cano

453381124

b)

TP 1°N → 2° entradas (1014)

TP 2°N → " " "

TP 2°N (0)

0	
1	
2	CÓDIGO
...	
1023	

TP 2°N (1)

0	
1	
...	
1023	

TP 2°N (1023)

0	
1	
...	
1023	

TP 1°N

0	
1	
...	
1023	

RAM	
0	TP 2°N (0)
1	
...	
1023	TP 2°N (1023)
1024	CÓDIGO
1025	DATOS 1
1026	DATOS 2
1027	PILA HEBRA
1028	P. MAIN 1
1029	MAIN 2
1030	MAIN 3
1031	MAIN 4

TAM  
PÁGINA: 4KB

Luego → código : 1 página

DATOS : 2 páginas

PILA HEBRA : 1 página

PILA MAIN : 4 páginas

Luego

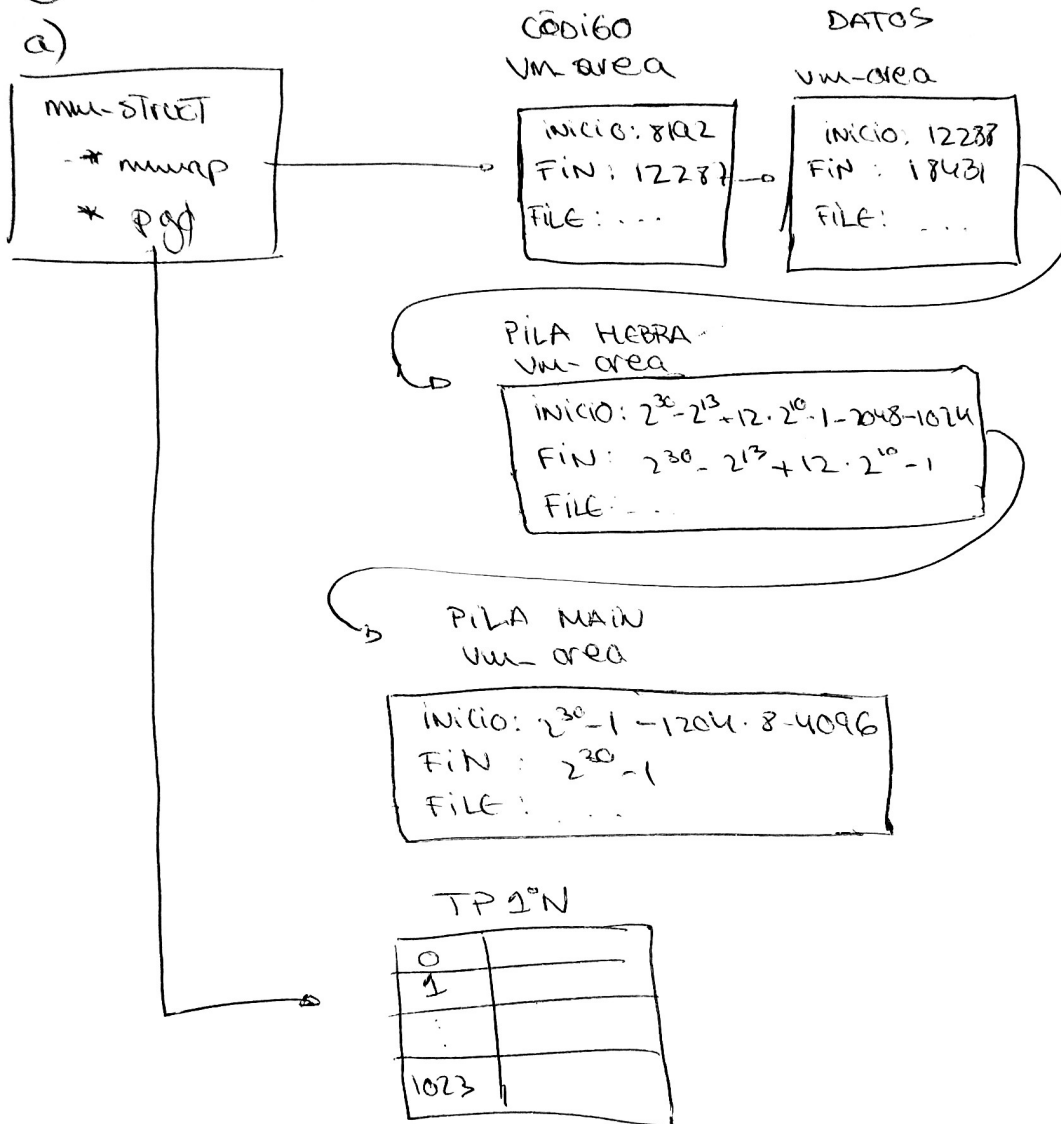
TP 2°N (0):

0	
1	
2	CÓDIGO
3	DATOS 1
4	DATOS 2
5	
...	
1023	

HABRÍA QUE SITUAR  
LA PILA HEBRA y LA PILA MAIN (NO ME DA TIEMPO).

②

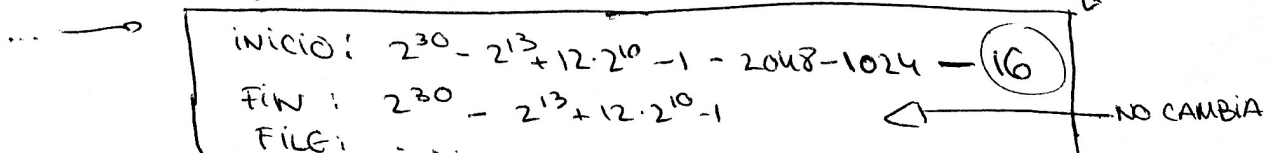
a)



b) Si la hebra empuja su pila 16 B eso corresponderá a aumentar el `Tau` de la pila 16 B luego el `vmu-area` de la pila de la hebra quedará:

PILA HEBRA  
VM-area

como la pila  
crece hacia posiciones  
de memoria menores (restamos)



# Tarea 4

Juan Valentin Guerrero Cano

453381124

K=8 L=1 M=1 N=2

3

a)

I1 = fd1 = open( ..., O\_RDWR );

I2 = fd2 = open( ..., O\_RDWR );

I3 = read( fd2, buffer2, M );

I4 = read( fd2, buffer2, M );

I5 = fork();

I6 = write( fd1, buffer2, 5 );

b)

DESCRIPTORES  
ARCHIVO

0	STDIN
1	STDOUT
2	STDERR
3	fd1
4	fd2

Hijo

0	STDIN
1	STDOUT
2	STDERR
3	fd1
4	fd2

TABLA  
ARCHIVOS ABIERTOS

CURSOS	REFERENCIAS	PERMISOS
10	2	RW-
28	2	RW-

Tabla  
dentro

datos	INOC
CSV	700
datos TXT	600

Tabla in-nodos

in-nodo	metadatos
700	...
600	...

Juan Valentin Guerrero Caro 453381124

④ FAT se basa en la asignación no contigua mediante enlazamiento.

En nuestra memoria tendremos una bloque que almacena nuestra Tabla FAT

EXT2 se basa en la asignación por indexación, es decir, índices que contendrán información acerca de los metadatos del archivo y los bloques donde se almacena

↳ Tabla de indexación combinada