COGNOMS:														
								ı						
NOM:														

IMPORTANTE leer atentamente antes de empezar el examen: Escriba los apellidos y el nombre antes de empezar el examen. Escriba un solo carácter por recuadro, en mayúsculas y lo más claramente posible. Es importante que no haya tachones ni borrones y que cada carácter quede enmarcado dentro de su recuadro sin llegar a tocar los bordes. Use un único cuadro en blanco para separar los apellidos y nombres compuestos si es el caso. No escriba fuera de los recuadros.

Problema 1. (5 puntos)

Dado el siguiente código escrito en C, que compilamos para un sistema linux de 32 bits:

Dibuja como quedarían almacenadas en memoria las estructuras s1 y s2, indicando claramente los

desplazamientos respecto al inicio, el tamaño de todos los campos y el tamaño de los structs.	

de la dirección.		

b) **Escribe** UNA ÚNICA INSTRUCCIÓN que permita mover **z.u[2].c** al registro **%ax**, siendo **z** una variable de tipo **s2** cuya dirección está almacenada en el registro **%ebx**. **Indica** claramente la expresión aritmética utilizada para el cálculo

Dado el siguiente código escrito en C:

```
int examen(short a, char b, int v[10], short M[4][4]) {
 int ii;
  short aa;
```

	<pre>short *matriz; int *vector;</pre>
	<pre>return v[ii];</pre>
	}
c)	Dibuja el bloque de activación de la rutina examen, indicando claramente los desplazamientos respecto a %ebp
	el tamaño de todos los campos.
d)	Traduce a ensamblador del x86 la instrucción return v[ii] USANDO EL MÍNIMO NÚMERO DE INSTRUCCIONE
~, —	y suponiendo que la subrutina ha usado los registros %eax, %ebx, %ecx y %edx.

COGNOMS:																									
NOM:]											
NOW.																									
Problema :	2. (5 p	unt	os)																						
Un sistema de cada ima una frecuer energía mu necesarios p	de vig agen d icia F o estra	iland ebe es su que	cia and realiz oficien la ru	arse ite p tina	en ara HFR	un tie proce (Hu	empo esar u	igua una ir	l o ir nage	nferio en en	or a ı T m	T. S nilis	se ha egui	a ca ndo	alcu s. L	lado Jn ar	que nálisi	un s de	proc rer	cesa ndin	ador nient	P fur	ncion consu	ando ımo (a de
Se ha decidido incorporar un acelerador hardware (AH) al sistema. AH realiza el procesado HFR vía hardware pero consumiendo la cuarta parte de la energía respecto la ejecución de la rutina en P. El resto del código, que llamaremos PNA (parte no acelerada) se sigue ejecutando en P y por tanto no se beneficia del acelerador.																									
a) Calcul a	a la ga	nand	cia en	ene	rgía	del si	stem	na cor	n ace	lerac	dor ((P+/	۱ (H	resp	oect	to al	siste	ma	P.						_
El acelerado que el tiem procesar ur	po de	eje	cuciór	n de	PNA	۹en	P es	prop	orcic	nal	al ti	emį	ро с	de d	ciclo	. Da	do d	que	no l	hay	ning	ún t	enef	icio e	en
acelerada ta		_						_	-						116	cuei	icia	ue r	ue	101	IIIIa	que	ia po	ii te i	10
b) Calcula																		ie po	odar	nos	redu	ıcir l	a fred	uenc	ia
de P a	la mit	ad si	n que	el ti	iemp	oo de	proc	esad	o de	una	ima	gen	sea	sup	peri	or a	Т.			—					
El procesado instrucciono														-						∍ P (ejecu	ıta 6	millo	nes	de
c) Calcula	a el CP	l y e	l tiem	рос	le ej	ecuci	ón e	n mili	segu	ndos	s (Te	exec	:) de	PN	IA e	n P.									
d) Calcula	a los N	1IPS	y MFI	OPS	de	PNA	en di	cho s	isten	na.															

El procesador P dispone de una cache con una política de escritura *Copy Back* y *Write Allocate*. En caso de acierto en la cache el tiempo de acceso es de 1 ciclo. En caso de fallo, hay una penalización de 60 ciclos para reemplazar un bloque NO modificado y de 120 ciclos para reemplazar un bloque modificado.

PNA realiza 8 millones de accesos a memoria y tiene una tasa de fallos (miss) del 10%. Sabemos que el 15% de los accesos son escrituras y que la probabilidad de que un bloque haya sido modificado en cache es del 20%.

e) C	alcula e	el tiempo	medio de acceso a memoria (Tma) en ciclos de la parte no acelerada (PNA).
Leer o	escribii	r un bloqu	e en memoria principal consume 100 nanoJoules (nJ).
f) C	alcula e	el consum	o total de energía de la memoria principal causada por los fallos de cache.
está co asociat	ompues tivo. La	sta por ur cache tie	direcciones lógicas de 36 bits y direcciones físicas de 24 bits. La jerarquía completa de memoria TLB, la memoria cache y la memoria principal. El TLB tiene 4 entradas y es completamente e un tamaño de 64 Kbytes, líneas de 64 bytes y es 4-asociativa. El tamaño de página del sistema e accede antes que la cache, por lo que la cache se indexa con direcciones físicas.
g) C	•	el número	de líneas, vías y conjuntos que tiene la cache. Especifica claramente cómo has realizado los
El nroc	esador	lanza un	acceso a la dirección lógica 0xEFABCD012 y sabemos que el contenido del TLB es:
VPN	,esado:	PPN	access a la all'ección logica oxer/lecesore y sabellos que el contellido del 125 es.
0xFA	BCD0	0xA00	
0xBC		0xB01	
0xEF/	ABCD	0xC02	
0xAB	CD01	0xD03	
-		-	cción física se accede, en qué conjunto de la cache se encuentra el dato y cuál es la etiqueta oria cache. Justifica la respuesta.