COGNO	MS:																										
NO	ом:																										
examen tachone	MPORTANTE 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 cachones 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. (2,5 puntos)																										
Problei	ma 1. ((2,5	pun	tos)																							
Un prog																										ota	ante.
	lcula e			_											-												
El proce																	volta	aje d	le 2	V y	tie	ne ı	una	carg	a ca	pac	citiva
b) Ca	Icula la	enei	rgía	cons	sum	ida a	al eje	cut	ar el	pro	ogra	ma	PS er	n el	prod	cesa	dor	C.									
Hemos	_					-				-													-				
	Icula e ograma					prod	cesad	lore	es id	ént	icos	a C	que	son	nec	cesa	arios	par	а сс	onse	guir	un	spe	•d-u	p de	5	en el
d) Ca	Icula Ic	s MF	LOF	'S de	PP	en d	licho	sist	tema	a pa	rale	lo (I	N pro	ces	ado	res	C).										
Una CPU B tiene disipada a la vez. procesa	un 75% a por lo . PS se o dor A y	6 del que ejecu 1 la pa	ren sim ta e arte	dimi ultár xclu para	ento near siva alela	o de nent men n en	A, pe te sol te so los 4	ero o p bre pro	con ued e el p oces	sum e fu oroc ado	ne so ncio esac res E	olo nar dor 3.	un 30 o un A. M	o% o pro ient	de la ocesa	a po ado	teno r A c	cia. L o los	a CI 4 pr	OCE:	stá sado	limi ores	tada B, p	por ero	r la p nunc	ote a t	encia odos
e) Ca	Icula la	gana	anci	a en	ene	rgía	de P	P re	espe	cto	PS e	en e	sta C	PU.													

COGNOMS:														
								1						
NOM:														

Problema 2. (2.5 puntos)

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

```
typedef struct {
                                          typedef struct {
```

		cabla[50]; ct int *h;
a)	a) Dibuja como quedarían almacenadas en memoria las estruc desplazamientos respecto al inicio, el tamaño de todos los campos y	
b)	b) Escribe UNA ÚNICA INSTRUCCIÓN en ensamblador que traduzca la una variable de tipo sb cuya dirección está almacenada en el re aritmética utilizada para el cálculo de la dirección.	

COGNOMS:														
								1						
NOM:														

Problema 3. (2.5 puntos)

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

```
int funcion(char x, short int y[3]) {
  int a;
  int *b;
  short int c[3];
  char d;
   . . .
  a = funcion(x, c) + *b;
   . . .
}
```

el tamaño de todos los campos.	<mark>%ebp</mark> у
er tamano de todos los campos.	

_			

b) Traduce a ensamblador x86 la sentencia a = funcion(x,c) + *b; que se encuentra en el interior de la subrutina, sabiendo que el valor de la expresión *b se encuentra en el registro %edx justo antes de la llamada. Se valorará usar el mínimo número de instrucciones y el mínimo número de accesos a memoria.

OMS:																											
NOM:																											
tiene dremo	un os u	CPI na t	idea tabla	al (s	supo e pá	onie igina	ndo as ur	que niniv	ca el c	da a que	es e	so a	me	mo	ria p	orin	cipa	l se	res	uelv	e en	1 (ciclo) de	e 8	ciclos	у
_ 1	mov.	1 \$: 1 \$:	1000 0, %	000 %ea	0, x	%ec	X		uta	en C),42 [,]	4s. L	as e	tiqu	etas	s "V'	'у"	bucl	e" e	stán	alin	eada	as a	1M	B:		
(dec.	1 %	есх		•																						
								os a	me	mo	ria p	orinc	ipal	pro	duci	idos	al e	ejecı	ıtar	el b	ucle.	¿Cı	uál e	es e	l tier	mpo d	de
- i£:			1-1-1			.l:	altina a				1					LD											
									-			-				-										ne qu	иe
				-													-			, خ در	iál es	s el	tam	año	mín	imo d	de
nente	deci	dim	os ut	tiliza	ar 2	cacl	hes (y 2 ⁻	TLBs	s) se	para	ados	de i	inst	rucc	ione	es y	dato	s. C	ada	TLB 1	tien	e 8 e	entr	adas	у сас	da
																	tos	4 as	ocia	itiva	. Nue	estro	sis [°]	tem	а со	n TLB	У
Cuánt	tos f	allos	s de	cacl	he d	lará	en to	otal	el b	ucle	ant	erio	r?														_
Cuánt	tos f	allos	s de '	TLB	dar	á en	tota	al el	buc	le a	nter	ior?															
	a inefi mos u Fenien LEB qu	ema 4. (2 nemos de tiene un dremos u ndremos ta ngamos el : mov mov ne: add dec jnz Calcula el nacceso a la remos usar ta remiendo e TLB que no	ema 4. (2.5 premos de un atiene un CPI dremos una atienemos tamb agamos el sigu movl \$ movl \$ decl \$ jnz bu. Calcula el númicoceso a la mero decidimos usar tienemos usar	ema 4. (2.5 puntinemos de un siste tiene un CPI ide dremos una tabla dremos también de movi \$1000 movi \$0, 30 deci %ecx jnz bucle. Calcula el número acceso a la memori deceso a la memori. Teniendo en cuent. TLB que nos permitira de la cacatiene un CPI ideal decidante. Cuántos fallos de cuent. Cuántos fallos de cuán	ema 4. (2.5 puntos) nemos de un sistema tiene un CPI ideal (dremos una tabla de idremos también que ngamos el siguiente có movl \$100000 movl \$0, %ea e: addl V(,%ecx decl %ecx jnz bucle Calcula el número de ideaceso a la memoria p TLB que nos permitiría Teniendo en cuenta qu TLB que nos permitiría Tenente decidimos utiliz 128 lineas. La cache o tiene un CPI ideal de ideaceso de cac	ema 4. (2.5 puntos) nemos de un sistema de tiene un CPI ideal (supo dremos una tabla de pá adremos también que no h ngamos el siguiente código mov1 \$1000000, mov1 \$0, %eax he: add1 V(,%ecx,4) dec1 %ecx jnz bucle Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria princi Calcula el número de acce deceso a la memoria prin	ema 4. (2.5 puntos) nemos de un sistema de cóm tiene un CPI ideal (suponie idremos una tabla de página idremos también que no hay fi ngamos el siguiente código qui mov1 \$1000000, %ec mov1 \$0, %eax e: addl V(,%ecx,4), % decl %ecx jnz bucle Calcula el número de accesos acceso a la memoria principal? Teniendo en cuenta que el TL TLB que nos permitiría tener el mente decidimos utilizar 2 cac 128 lineas. La cache de instru tiene un CPI ideal de 6 ciclos. Cuántos fallos de cache dará	ema 4. (2.5 puntos) nemos de un sistema de cómputo tiene un CPI ideal (suponiendo dremos una tabla de páginas un idremos también que no hay fallos agamos el siguiente código que se mov1 \$1000000, %ecx mov1 \$0, %eax dec1 %ecx jnz bucle Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal? Calcula el número de accesos físicaceso a la memoria principal?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que tiene un CPI ideal (suponiendo que dremos una tabla de páginas uninivadremos también que no hay fallos de agamos el siguiente código que se ejecto mov1 \$1000000, %ecx mov1 \$0, %eax e: addl V(,%ecx,4), %eax decl %ecx jnz bucle Calcula el número de accesos físicos a acceso a la memoria principal? Calcula el número de accesos físicos a acceso a la memoria principal? Teniendo en cuenta que el TLB es com TLB que nos permitiría tener el máximo mente decidimos utilizar 2 caches (y 2 de 128 lineas. La cache de instrucciones tiene un CPI ideal de 6 ciclos. Las págio de 128 lineas de 130 de cache dará en total el cuántos fallos el cuántos fallos el cuántos fallos el cuántos el	ema 4. (2.5 puntos) nemos de un sistema de cómputo que ut tiene un CPI ideal (suponiendo que cardremos una tabla de páginas uninivel o dremos también que no hay fallos de páginas des es ejecuta mov1 \$1000000, %ecx mov1 \$0, %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a meroceso a la memoria principal? Calcula el número de accesos físicos a meroceso a la memoria principal? Teniendo en cuenta que el TLB es completa que nos permitiría tener el máximo número de acceso de completa que nos permitiría tener el máximo número de acceso de completa que nos permitiría tener el máximo número de acceso de completa que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de completa que el TLB que nos permitiría tener el máximo número de comple	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza tiene un CPI ideal (suponiendo que cada a dremos una tabla de páginas uninivel que ndremos también que no hay fallos de página. ngamos el siguiente código que se ejecuta en o mov1 \$1000000, %ecx mov1 \$0, %eax ne: add1 V(,%ecx,4), %eax dec1 %ecx jnz buc1e Calcula el número de accesos físicos a memo neceso a la memoria principal? Teniendo en cuenta que el TLB es completam TLB que nos permitiría tener el máximo númer mente decidimos utilizar 2 caches (y 2 TLBs) se 128 lineas. La cache de instrucciones es de a tiene un CPI ideal de 6 ciclos. Las páginas del Cuántos fallos de cache dará en total el bucle Cuántos fallos de cache dará en total el bucle	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza me tiene un CPI ideal (suponiendo que cada acce dremos una tabla de páginas uninivel que es adremos también que no hay fallos de página. negamos el siguiente código que se ejecuta en 0,42 mov1 \$1000000, %ecx mov1 \$0, %eax ne: add1 V(,%ecx,4), %eax dec1 %ecx jnz buc1e Calcula el número de accesos físicos a memoria pa neceso a la memoria principal? Teniendo en cuenta que el TLB es completamente TLB que nos permitiría tener el máximo número de nente decidimos utilizar 2 caches (y 2 TLBs) separa 128 lineas. La cache de instrucciones es de acces tiene un CPI ideal de 6 ciclos. Las páginas del siste Cuántos fallos de cache dará en total el bucle ant	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memor tiene un CPI ideal (suponiendo que cada acceso a dremos una tabla de páginas uninivel que es capa dremos también que no hay fallos de página. ngamos el siguiente código que se ejecuta en 0,424s. L mov1 \$1000000, %ecx mov1 \$0, %eax e: add1 V(,%ecx,4), %eax dec1 %ecx jnz buc1e Calcula el número de accesos físicos a memoria princ acceso a la memoria principal? Teniendo en cuenta que el TLB es completamente asc TLB que nos permitiría tener el máximo número de aci nente decidimos utilizar 2 caches (y 2 TLBs) separados 128 lineas. La cache de instrucciones es de acceso di tiene un CPI ideal de 6 ciclos. Las páginas del sistema	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria v tiene un CPI ideal (suponiendo que cada acceso a me dremos una tabla de páginas uninivel que es capaz de idremos también que no hay fallos de página. ngamos el siguiente código que se ejecuta en 0,424s. Las e mov1 \$1000000, %ecx mov1 \$0, %eax ne: add1 V(,%ecx,4), %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal acceso a la memoria principal? Calcula el número de accesos físicos a memoria principal acceso a la memoria principal? TEB que nos permitiría tener el máximo número de acierto nente decidimos utilizar 2 caches (y 2 TLBs) separados de 128 lineas. La cache de instrucciones es de acceso direct tiene un CPI ideal de 6 ciclos. Las páginas del sistema son Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtu tiene un CPI ideal (suponiendo que cada acceso a memo dremos una tabla de páginas uninivel que es capaz de re idremos también que no hay fallos de página. ngamos el siguiente código que se ejecuta en 0,424s. Las etique mov1 \$10000000, %ecx mov1 \$0, %eax ne: add1 V(,%ecx,4), %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal pro deceso a la memoria principal? Calcula el número de accesos físicos a memoria principal pro deceso a la memoria principal? Tenendo en cuenta que el TLB es completamente asociativo TLB que nos permitiría tener el máximo número de aciertos er mente decidimos utilizar 2 caches (y 2 TLBs) separados de inst 128 lineas. La cache de instrucciones es de acceso directo y tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de accenta de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual si tiene un CPI ideal (suponiendo que cada acceso a memoria pi dremos una tabla de páginas uninivel que es capaz de resolv idremos también que no hay fallos de página. negamos el siguiente código que se ejecuta en 0,424s. Las etiquetas mov1 \$1000000, %ecx mov1 \$0, %eax ne: add1 v(,%ecx,4), %eax dec1 %ecx jnz buc1e Calcula el número de accesos físicos a memoria principal product acceso a la memoria principal? Calcula el número de accesos físicos a memoria principal product acceso a la memoria principal? Teniendo en cuenta que el TLB es completamente asociativo con TLB que nos permitiría tener el máximo número de aciertos en el tene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin m tiene un CPI ideal (suponiendo que cada acceso a memoria prin dremos una tabla de páginas uninivel que es capaz de resolver d idremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "Vi mov1 \$1000000, %ecx mov1 \$0, %eax e: addl V(,%ecx,4), %eax decl %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos acceso a la memoria principal? a ineficiencia del sistema decidimos incorporar al procesador un TLB y mos usar tiene líneas de 32 bytes. Tanto el TLB como la cache son uni feniendo en cuenta que el TLB es completamente asociativo con ree TLB que nos permitiría tener el máximo número de aciertos en el bucl mente decidimos utilizar 2 caches (y 2 TLBs) separados de instruccione 128 lineas. La cache de instrucciones es de acceso directo y la de da tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB. Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memor tiene un CPI ideal (suponiendo que cada acceso a memoria principa dremos una tabla de páginas uninivel que es capaz de resolver toda idremos también que no hay fallos de página. Ngamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y " mov1 \$1000000, %ecx mov1 \$0, %eax .e: add1 V(,%ecx,4), %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al esceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al esceso a la memoria principal? Teniendo en cuenta que el TLB es completamente asociativo con reempl TLB que nos permitiría tener el máximo número de aciertos en el bucle an 128 lineas. La cache de instrucciones es de acceso directo y la de datos tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB. Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria tiene un CPI ideal (suponiendo que cada acceso a memoria principal se dremos una tabla de páginas uninivel que es capaz de resolver todas la idremos también que no hay fallos de página. Ingamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "buclimov1 \$1000000, %ecx mov1 \$0, %eax Le: add1 V(,%ecx,4), %eax dec1 %ecx Jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de accesos físicos a memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de accesos físicos a memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de accesos físicos a memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de accesos físicos a memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de accesos a la memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de acceso a la memoria principal producidos al ejectoceso a la memoria principal? Talcula el número de acceso a la memoria principal producidos al ejectoceso a la memoria principal?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria caci tiene un CPI ideal (suponiendo que cada acceso a memoria principal se res dremos una tabla de páginas uninivel que es capaz de resolver todas las tr dremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" e mov1 \$1000000, %ecx mov1 \$1000000, %ecx mov1 \$0, %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar acceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar acceso a la memoria principal? Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Tenente decidimos utilizar 2 caches (y 2 TLBs) separados de instrucciones y datos. C 128 lineas. La cache de instrucciones es de acceso directo y la de datos 4 asocia tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB. Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelv dremos una tabla de páginas uninivel que es capaz de resolver todas las traduc dremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están movl \$1000000, %ecx movl \$0, %eax .e: addl V(,%ecx,4), %eax decl %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el b acceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el b acceso a la memoria principal? Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cu TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cu TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Tenente decidimos utilizar 2 caches (y 2 TLBs) separados de instrucciones y datos. Cada 128 lineas. La cache de instrucciones es de acceso directo y la de datos 4 asociativa tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB. Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni TLB tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelve en dremos una tabla de páginas uninivel que es capaz de resolver todas las traduccion idremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están alim mov! \$1000000, %ecx mov! \$0, %eax e: add! V(,%ecx,4), %eax dec! %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. Inceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. Inceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. Inceso a la memoria principal?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni TLB. El tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelve en 1 d'dremos una tabla de páginas uninivel que es capaz de resolver todas las traducciones o didremos también que no hay fallos de página. ngamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están alinead: mov1 \$1000000, %ecx mov1 \$0, %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Co caceso a la memoria principal? a ineficiencia del sistema decidimos incorporar al procesador un TLB y una memoria cache. La mos usar tiene líneas de 32 bytes. Tanto el TLB como la cache son unificados para instruccione feniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cuál es el TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Tenente decidimos utilizar 2 caches (y 2 TLBs) separados de instrucciones y datos. Cada TLB tiene 128 lineas. La cache de instrucciones es de acceso directo y la de datos 4 asociativa. Nuestro tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB. Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni TLB. El siste tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelve en 1 ciclo dremos una tabla de páginas uninivel que es capaz de resolver todas las traducciones en u dremos también que no hay fallos de página. ngamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están alineadas a mov1 \$1000000, %ecx mov1 \$0,0 %ecx mov1 \$0,0 %ecx mov1 \$0,0 %ecx mov1 \$0,0 %ecx jnz bucle nece: add1 V(,%ecx,4), %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál el acceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál el acceso a la memoria principal? Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cuál es el tam TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cuál es el tam TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni TLB. El sistema tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelve en 1 ciclo) di dremos una tabla de páginas uninivel que es capaz de resolver todas las traducciones en un sidremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están alineadas a 1M mov1 \$1000000, \$ecx mov1 \$0,0 \$eax mov1 \$0,0 \$eax decl \$ecx jnz bucle Le: add1 V(,\$ecx,4), \$eax decl \$ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál es elecceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos para instrucciones y dato enceso a la memoria principal? Tale que nos permitiría tener el máximo número de aciertos en el bucle anterior? Tale que nos permitiría tener el máximo número de aciertos en el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni TLB. El sistema fun tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelve en 1 ciclo) de 8 d dremos una tabla de páginas uninivel que es capaz de resolver todas las traducciones en un solo idremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están alineadas a 1MB: mov1 \$1000000, %ecx mov1 \$0, %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál es el tier caceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál es el tier caceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál es el tier caceso a la memoria principal? Calcula el número de accesos físicos a memoria cache son unificados para instrucciones y datos. Ceniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cuál es el tamaño mín LB que nos permitiría tener el máximo número de aciertos en el bucle anterior? Tenente decidimos utilizar 2 caches (y 2 TLBs) separados de instrucciones y datos. Cada TLB tiene 8 entradas 128 lineas. La cache de instrucciones es de acceso directo y la de datos 4 asociativa. Nuestro sistema co tiene un CPI ideal de 6 ciclos. Las páginas del sistema son de 4 KB. Cuántos fallos de cache dará en total el bucle anterior?	ema 4. (2.5 puntos) nemos de un sistema de cómputo que utiliza memoria virtual sin memoria cache ni TLB. El sistema funciona tiene un CPI ideal (suponiendo que cada acceso a memoria principal se resuelve en 1 ciclo) de 8 ciclos dremos una tabla de páginas uninivel que es capaz de resolver todas las traducciones en un solo acces idremos también que no hay fallos de página. Igamos el siguiente código que se ejecuta en 0,424s. Las etiquetas "V" y "bucle" están alineadas a 1MB: mov1 \$10,00000, %ecx mov1 \$0, %ecx, 4), %eax dec1 %ecx jnz bucle Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál es el tiempo de caceso a la memoria principal? Calcula el número de accesos físicos a memoria principal producidos al ejecutar el bucle. ¿Cuál es el tiempo de caceso a la memoria principal? ai ineficiencia del sistema decidimos incorporar al procesador un TLB y una memoria cache. La memoria cache que mos usar tiene líneas de 32 bytes. Tanto el TLB como la cache son unificados para instrucciones y datos. Teniendo en cuenta que el TLB es completamente asociativo con reemplazo LRU, ¿cuál es el tamaño mínimo o TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior? TLB que nos permitiría tener el máximo número de aciertos en el bucle anterior?