

Arquitecturas y Organización de Computadoras I - Segundo Parcial 10 de noviembre de 2023.



Nombre y Apellido LEGAJO LEGAJO

Ejercicio 1. Desarrollar un programa en lenguaje ensamblador MIPS para extraer el valor de los tres números, y almacenar la extracción en el espacio reservado a partir de la etiqueta "valores".

Ejemplo: si en mensaje se encuentra el texto "los 2 sistemas contienen 9 archivos y 3 directorios", el programa debe extraer los valores 2, 9 y 3; y almacenar cada uno (su valor numérico, no su representación texto) en el espacio reservado en "valores". Tenga en cuenta que el mensaje puede variar, y por supuesto, los números en el mensaje también. Los números que existan en mensaje tienen un rango [0, 9].

Utilice el siguiente código en ensamblador como referencia para realizar el ejercicio.

```
.data
memoria:
valores:
    .word 0, 0, 0
mensaje:
    .asciiz "El resultado de la suma de los 3 números es 7, y es múltiplo de 2"
    .text
    .globl main
    .globl __start

__start:
main:
    # COMPLETAR

# retorna al SO
    move $a0, $0
    li $v0, 4001
    syscall
```

Ejercicio 2. Complete la subrutina siguiendo la convención de llamadas a procedimiento. # esta subrutina obtiene un argumento N y devuelve 0 si el argumento N es impar y 1 si el argumento N es par. es_par:

move \$s3,	# colocar en \$s3 el valor del argumento N de entrada a la función
	# ahora utilizaremos la instrucción "and" para saber si \$s3 es par o impar
move \$t3, \$s3	# colocamos el resultado (si es par o impar) en \$t3
jal mostrar_por_pantalla move, \$t3	# colocamos en el valor a retornar a la función llamadora

jr # COMPLETAR



Arquitecturas y Organización de Computadoras I - Segundo Parcial 10 de noviembre de 2023.



Nombre y Apellido LEGAJO LEGAJO

Ejercicio 3.

```
.data
memoria:
solsticio:
               .byte 1, 2, 3
              .word 8
entrada·
              .asciiz "Habian nueve caramelos en el frasco."
caramelos:
password:
              .word 0
               .asciiz "El formato especial."
formato:
prom:
               .word 0
        .text
        .globl main
        .globl start
main:
       li $t0, 0
       li $t5, 0
loop:
       lb $t3, formato($t0)
       lb $t4, caramelos($t0)
       subu $t5, $t5, $t3
       addu $t5, $t5, $t4
       addi $t0, $t0, 1
       bne $t3, $zero, loop:
       sw $t5, prom
# retorna al SO
       move $a0, $0
       li
               $v0, 4001
        syscall
```

Este programa se ejecuta en una computadora con procesador MIPS de 100 khz (k significa mil). En el mismo se cuenta con una cache de mapeo directo de 8 entradas. En cada entrada caben 2 palabras. Cada acceso a caché demora 10us (microsegundos). Cada acceso a memoria principal demora 80us. El ensamblador traduce pseudoinstrucciones a 2 instrucciones máquina. Se realizó una acción de anular la caché, por lo que todos los bits de validez de las entradas están en cero. El segmento de datos se almacena en la dirección 0x444000.

- a. ¿Cuánto demora en ejecutarse este programa si se tiene en cuenta la caché?
- b. ¿Cuanto demora en ejecutarse este programa si el 100% de los accesos a memoria hubiesen sido fallos?
- c. ¿Cuál es el tiempo de acceso medio?
- d. ¿Cuánto ocupa este programa en memoria?
- e. Indique en qué entrada de la caché se almacena:
 - la última letra e en "nueve" (de caramelos)
 - la primera letra e de "especial" (de formato)

Dec	ec Hex		Dec He		Hex	Dec		Hex	Dec 1		Hex	Dec		Hex	x Dec		Hex	Hex Dec		Hex Dec Hex		
0	00 N	UL 1	6 :	10	DLE	32	20		48	30	0	64	40	@	80	50	P	96	60	112	70 p	
1	01 S	OH 1	7 :	11	DC1	33	21	!	49	31	1	65	41	A	81	51	Q	97	61 8	a 113	71 q	
2	02 S	TX 1	8 :	12	DC2	34	22		50	32	2	66	42	В	82	52	R	98	62 1	b 114	72 r	
3	03 E	TX 1:	9 :	13	DC3	35	23	#	51	33	3	67	43	C	83	53	S	99	63 (c 115	73 s	
4	04 E	OT 2	0 :	14	DC4	36	24	\$	52	34	4	68	44	D	84	54	T	100	64 d	116 7	4 t	
5	05 E	NQ 2	1 :	15	NAK	37	25	%	53	35	5	69	45	E	85	55	U	101	65 e	117 7	5 u	
6	06 A	CK 2	2 :	16	SYN	38	26	&	54	36	6	70	46	F	86	56	v	102	66 £	118 7	6 v	
7	07 B	BEL 2	3 :	17	ETB	39	27	•	55	37	7	71	47	G	87	57	W	103	67 g	119 7	7 w	
8	08 B	S 2	4 :	18	CAN	40	28	(56	38	8	72	48	H	88	58	X	104	68 h	120 7	8 x	
9	09 H	T 2	5 3	19	EM	41	29)	57	39	9	73	49	I	89	59	Y	105	69 i	121	79 y	
10	OA L	F 2	6 :	1A	SUB	42	2A	*	58	3 A	:	74	4A	J	90	5A	Z	106	6A j	122 7	Αz	
11	OB V	T 2	7 :	1B	ESC	43	2B	+	59	3B	;	75	4B	K	91	5B	[107	6B k	123 7	ъ {	
12	OC F	F 2	8 :	1C	FS	44	2C	,	60	3 C	<	76	4C	L	92	5C	\	108 6	C 1	124 7	C	
13	OD C	R 2	9 :	1D	GS	45	2D	-	61	3D	=	77	4D	M	93	5D]	109 6	D m	125 7	D }	
14	OE S	30	0 :	1E	RS	46	2E		62	3E	>	78	4E	N	94	5E	^	110 6	Εn	126 7	E ~	
15	OF S	SI 3:	1 :	1F	US	47	2 F	/	63	3 F	?	79	4 F	0	95	5 F	_	111 6	Fο	127 7	F DEL	
32	32 20 : espacio en blanco																					