ORGA1: Arquitectura y Seguimiento

Organización del Computador I

Verónica Coy

Departamento de Computación - FCEyN UBA

Primer Cuatrimestre 2018 - Turno Mañana

Programación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierr

Ciclo de Vida de un Programa

Ciclo de Vida de un Programa

Programación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierro

Ciclo de Vida de un Programa

Ciclo de Vida de un Programa

Programación: Escribir un algoritmo en lenguaje ensamblador (o assembly). Esto es el código fuente. Programación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierro

Ciclo de Vida de un Programa

Ciclo de Vida de un Programa

- Programación: Escribir un algoritmo en lenguaje ensamblador (o assembly). Esto es el código fuente.
- Ensamblado: Un programa llamado Ensamblador toma el código fuente y lo traduce a un código máquina.

Ciclo de Vida de un Programa

- Programación: Escribir un algoritmo en lenguaje ensamblador (o assembly). Esto es el código fuente.
- Ensamblado: Un programa llamado Ensamblador toma el código fuente y lo traduce a un código máquina.
 - Resuelve las directivas dirigidas al **Ensamblador**.
 - Calcula el tamaño de cada instrucciones, y resuelve las etiquetas.
 - Traduce las instrucciones a 0s y 1s.

Programación

- Programación: Escribir un algoritmo en lenguaje ensamblador (o assembly). Esto es el código fuente.
- Ensamblado: Un programa llamado Ensamblador toma el código fuente y lo traduce a un código máquina.
 - Resuelve las directivas dirigidas al **Ensamblador**.
 - Qualcula el tamaño de cada instrucciones, y resuelve las etiquetas.
 - 3 Traduce las instrucciones a 0s y 1s.
- Carga: Se le indica al Ensamblador una dirección inicial, y éste copia el código máquina en la Memoria desde esa posición en adelante. Luego, carga esa misma dirección en el PC (Program Counter).

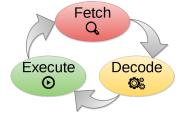
Ciclo de Vida de un Programa

- Programación: Escribir un algoritmo en lenguaje ensamblador (o assembly). Esto es el código fuente.
- Ensamblado: Un programa llamado Ensamblador toma el código fuente y lo traduce a un código máquina.
 - Resuelve las directivas dirigidas al **Ensamblador**.
 - Calcula el tamaño de cada instrucciones, y resuelve las etiquetas.
 - 3 Traduce las instrucciones a 0s y 1s.
- Carga: Se le indica al Ensamblador una dirección inicial, y éste copia el código máquina en la Memoria desde esa posición en adelante. Luego, carga esa misma dirección en el PC (Program Counter).
- Ejecución: El CPU da inicio a su ciclo de ejecución comenzando por la posición indicada en el PC.

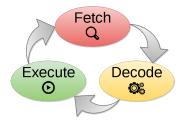
Programación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierr

Ciclo de Ejecución

Ciclo de Ejecución



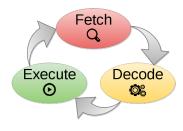
Ciclo de Ejecución



Fetch La UC (Unidad de Control) obtiene una instrucción de la posición de Memoria a la que apunta el PC y lo incrementa (Si es necesario:

busca más palabras de la instrucción usando el **PC** e incrementándolo cada vez.)

Ciclo de Ejecución



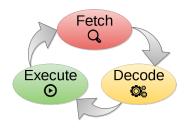
Fetch La UC (Unidad de Control) obtiene una instrucción de la posición de Memoria a la que apunta el PC y lo incrementa (Si es necesario: busca más palabras de la instrucción

usando el PC e incrementándolo cada

Decode La UC decodifica la instrucción.

vez.)

Ciclo de Ejecución



Fetch La UC (Unidad de Control) obtiene una instrucción de la posición de Memoria a la que

apunta el PC

y lo incrementa (Si es necesario: busca más palabras de la instrucción usando el PC e incrementándolo cada vez.)

Decode La UC decodifica la instrucción.

Execute La **UC** ejecuta la instrucción.

Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

MOV R1,[[et1]] main:

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004 Tenemos que:

Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

Tenemos que:

 ver cuántas palabras necesita cada instrucción

Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

Tenemos que:

- ver cuántas palabras necesita cada instrucción
- calcular los valores de las etiquetas

Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

Tenemos que:

- ver cuántas palabras necesita cada instrucción
- calcular los valores de las etiquetas

DW (Define Word): directiva al ensamblador que provoca que en la posición de memoria que le corresponde, aparezca el valor indicado.

dirección ocupa

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

main: MOV R1, [[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

 0×0000 dos palabras

0x0002

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

0x0004

Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

 0×0004 dos palabras

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

0x0004 dos palabras

0×0006

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

0x0004 dos palabras

0x0006 una palabra

main: MOV R1,[[et1]]

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

0x0004 dos palabras

0x0006 una palabra

0×0007

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

main:	MOV	R1.	[[et1]]
mam.	IVIOV	ι\⊥,	1166711

ADD [R1],0x6000

JMP main

et1: DW 0x0007

et2: DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra

una palabra

0×0007

		dirección	ocupa
main:	MOV R1,[[et1]]	0×0000	dos palabras
	ADD [R1],0x6000	0×0002	dos palabras
	JMP main	0×0004	dos palabras
et1:	DW 0x0007	0×0006	una palabra
et2:	DW 0x0004	0×0007	una palabra

La etiqueta main corresponde a la dirección de memoria 0x0000. La etiqueta et1 corresponde a la dirección de memoria 0x0006. La etiqueta et2 corresponde a la dirección de memoria 0x0007.

Reemplazando las etiquetas por valores de las direcciones estamos listos para codificar.

MOV R1,[[0×0006]]
ADD [R1],0×6000
JMP 0×0000
DW 0×0007
DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0×0006]]
ADD [R1],0×6000
JMP 0×0000
DW 0×0007
DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0×0006]]
ADD [R1],0x6000
JMP 0×0000
DW 0×0007
DW 0×0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

	0000	0001	0002	0003	0004	0005	0006	0007
ı								



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0×0006]]
ADD [R1],0×6000
JMP 0x0000
DW 0×0007
DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006						



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0x0006]]
ADD [R1],0×6000
JMP 0×0000
DW 0×0007
DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006						



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

	MOV R1,[[0x0006]]
	ADD [R1],0x6000
	JMP 0×0000
	DW 0×0007
	DW 0×0004
۰	

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000				



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0×0006]]
ADD [R1],0x6000
JMP 0×0000
DW 0x0007
DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000				



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0x0006]]
ADD [R1],0x6000
JMP 0×0000
DW 0×0007
DW 0×0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000		



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0×0006]]
ADD [R1],0×6000
JMP 0x0000
DW 0×0007
DW 0×0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000		



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0x0006]]
ADD [R1],0x6000
JMP 0×0000
DW 0×0007
DW 0×0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000	0007	



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

MOV R1,[[0×0006]]
ADD [R1],0×6000
JMP 0x0000
DW 0x0007
DW 0x0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000	0007	



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

ADD [R1],0x6000 JMP 0x0000
JMP 0×0000
DW 0×0007
DW 0×0004

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	dos palabras
0×0006	una palabra
0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000	0007	0004



Codifiquemos el siguiente programa y carguémoslo desde la posición de memoria 0x0000:

	dirección	ocupa
MOV R1,[[0×0006]]	0×0000	dos palabras
ADD [R1],0x6000	0×0002	dos palabras
JMP 0×0000	0×0004	dos palabras
DW 0×0007	0×0006	una palabra
DW 0×0004	0×0007	una palabra

Traducidas las etiquetas lo codificamos y cargamos en memoria:

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000	0007	0004



Empezamos a hacer el ciclo **Fetch-Decode-Execute** con el **PC** = 0x0000.

0000	0001	0002	0003	0004	0005	0006	0007
1858	0006	2C40	6000	A000	0000	0007	0004

Valor	es F	C	SP		R0	R1	F	2	R3	R4	R5	R6	R7		Ζ	С	V	N
iniciales		0000	FFEF		0000	00	00 C	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	l +	-3	+4	+5	+6	+7	+8	+9	+A	+B	+C	T +	D	+E	+F
0000	1858	0006	2C40	60	000 A	4000	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	0000	00	000 0	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra		Inst deco		
1													٦
	Ejecución						•				Flags		
2													٦
	Ejecución						•				Flags		٦
3													٦
	Ejecución										Flags		٦

Valor	es F	C	SP		R0	R1	F	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF		0000	00	00 C	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+	3 -	+4	+5	+6	+7	+8	+9	+A	+B	+C	+1	D	+E	+F
0000	1858	0006	2C40	600	00 A	000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	000	00 00	000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000										
	Ejecución						•				Flags
2											
	Ejecución						•				Flags
3											
	Ejecución										Flags

Valor	es F	C	SP	F	R0	R1	F	R2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les (0000	FFEF	0	0000	000	00 C	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+	-4	+5	+6	+7	+8	+9	+A	+B	+C	+1	D	+E	+F
0000	1858	0006	2C40	6000	0 AC	000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0000	0 00	00	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instru decod	icción ificada	
1	0000	FFEF	0000										
	Ejecución						•				Flags		\Box
2													
	Ejecución						•				Flags		\Box
3													
	Ejecución										Flags		

Valor	es F	C	SP	R	0	R1	R:	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF	0	000	0000	00	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+4	+	5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6000) A00	0 00	00	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0000	0000	000	00	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Flags			
1	0000	FFEF	0000	1858										٦
	Ejecución						•				Flags			٦
2														٦
	Ejecución						•				Flags			٦
3														٦
	Ejecución										Flags			٦

Valor	es F	C	SP	R	0 R	1	R2	R3	R4	R5	R6	R7		Z C	V	N
inicia	les (000	FFEF	00	000 00	000	0000	0000	0000	0000	0000	000	00	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	1858	0006	2C40	6000	A000	0000	0007	0004	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000						
	Ejecución						•				Flags
2											
	Ejecución						•				Flags
3											
	Ejecución										Flags

Valor	es F	C	SP	R	0	R1	R:	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF	0	000	0000	00	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+4	+	5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6000) A00	0 00	00	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0000	0000	000	00	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001					
	Ejecución			•	•		•				Flags
2											
	Ejecución				•		•				Flags
3											
	Ejecución										Flags

Valor	es F	C	SP		R0	R1	F	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF		0000	00	00 C	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+	3 -	+4	+5	+6	+7	+8	+9	+A	+B	+C	+1	D	+E	+F
0000	1858	0006	2C40	600	00 A	000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	000	00 00	000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			
	Ejecución						•				Flags
2											
	Ejecución						•				Flags
3											
	Ejecución										Flags

Valor	es F	C	SP	R	0	R1	R:	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF	0	000	0000	00	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+4	+	5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6000) A00	0 00	00	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0000	0000	000	00	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución										Flags
2											
	Ejecución										Flags
3											
	Ejecución										Flags

Valor	es F	PC	SP	R	0	R1	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les 0	0000	FFEF	0	000	0000	0000	0000	0000	0000	0000	000	00	0	0	0	0
+0														+F			
0000	1858	0006	2C40	6000	A00	000	00 00	000	4 0000	0000	0000	0000	0000	00	00	0000	0000
	0000	0000	0000	0000	0000	000	00 00	000	0000	0000	0000	0000	0000	00	20	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2											
	Ejecución										Flags
3											
	Ejecución										Flags

Valor	es	PC	SP		R0	R1		R2	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les	0000	FFEF		0000	00	00 (0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+	-3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	,	+E	+F
0000	1858	0006	2C40	600	00 A	4000	0000	0007	0004	0000	0000	0000	0000	0000	000	0	0000	0000
0010	0000	0000	0000	000	00 C	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción
							palabra		palabra		decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0)×0006]]=[0×0007]	=4						Flags
2	0002										
	Ejecución										Flags
3											
	Ejecución				•						Flags

Valor	es	PC	SP		R0	R1		R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les	0000	FFEF		0000	00	00 (0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+	3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	1858	0006	2C40	600	00 A	000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	000	00 0	000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4		•				Flags
2	0002			2C40							
	Ejecución						•				Flags
3											
	Ejecución										Flags

Valor	es l	PC	SP		R0	R1	. F	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF		0000	00	00 0	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+	-3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	1858	0006	2C40	600	00 /	A000	0000	0007	0004	0000	0000	0000	0000	0000	000	0	0000	0000
0010	0000	0000	0000	000	00 (0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción
							palabra		palabra		decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	x0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000						
	Ejecución										Flags
3											
	Ejecución				•						Flags

Valor	es	PC	SP	F	R0	R1	R	2	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les	0000	FFEF	(0000	000	0 0	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	3 +	4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	600	0 A0	00	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	0000	000	0 000	00	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003					
	Ejecución										Flags
3											
	Ejecución										Flags

Valor	es l	PC	SP		R0	R1	.	R2	R3	R4	R5	R6	R7		Z	С	V	N
inicia	les (0000	FFEF		0000	0 00	00	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	-	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	-		A000	0000	0007	0004	0000	0000	0000	0000	0000	00	_	0000	0000
0010	0000	0000	0000	00	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	x0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			
	Ejecución										Flags
3											
	Ejecución										Flags

Valor	es F	PC	SP	RO) R:	1 F	R2	R3	R4	R5	R6	R7		Z 0	C V	N
inicia	les (0000	FFEF	00	00 00	000	0000	0000	0000	0000	0000	000	00	0 (0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	1858	0006	2C40	6000	A000	0000	0007	0004	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción
	1.0	51	[31 +1]	111	mistruccion - 1er parabra	1.0	palabra	1 0	palabra	1 C	decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	x0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución										Flags
3											
	Ejecución										Flags

Valor	es F	C	SP	F	R0	R1	R	2	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les 0	0000	FFEF	C	0000	000	0 0	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+	4	+5	+6	+7	+8	+9	+A	+B	+C	+1	D	+E	+F
0000	1858	0006	2C40	600	0 A0	00	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	000	0 000	00 (0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	РС	Instrucción decodificada
							1		рагарга		
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0×0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0x0004	4]=0×A00	0+0×60	00=0×0000						Flags
3											
	Ejecución										Flags

Valor	es l	PC	SP		R0	R1	. F	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF		0000	00	00 C	000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	1858	0006	2C40	600	0 00	0000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	000	0 00	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002	parabra		MOV R1, [[0x0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0×000	4]=0×A00	0+0×600	00=0×0000						Flags 1 1 0 0
3											
	Ejecución										Flags

Valor	es l	PC	SP		R0	R1	1	R2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les (0000	FFEF		000	0 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6	000	0000	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	0000	0	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC		struc codifi		
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV	₹1, [[0×0	006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags			
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0	×60	00
	Ejecución	[0×000	4]=0×A00	0+0×60	00=0×0000						Flags	1	1	0 0
3	0004													
	Ejecución										Flags			

Valor	es F	C	SP	R) R	1 F	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF	00	00 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6000	0000	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	x0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0×000	4]=0×A00	0+0×600	00=0×0000						Flags 1 1 0 0
3	0004			0000							
	Ejecución										Flags

Valor	es l	PC	SP		R0	R	L	R2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les (0000	FFE	F	000	00 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+	-2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	5 2C	40	6000	0000	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	000	00	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	x0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0×000	4]=0×A00	0+0×600	00=0×0000						Flags 1 1 0 0
3	0004			0000	0000 0000 0000 0000						
	Ejecución										Flags

Valor	es l	PC	SP		R0	R1	L	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF		000	0 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6	000	0000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0x000	4]=0×A00	0+0×600	00=0×0000						Flags 1 1 0 0
3	0004			0000	0000 0000 0000 0000						
	Ejecución										Flags

Valor	es l	PC	SP		R0	R1	L	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF		000	0 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6	000	0000	0000	0007	0004	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002	percent		MOV R1, [[0×0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0x0004	4]=0×A00	0+0×600	00=0×0000						Flags 1 1 0 0
3	0004			0000	0000 0000 0000 0000	0005					
	Ejecución										Flags

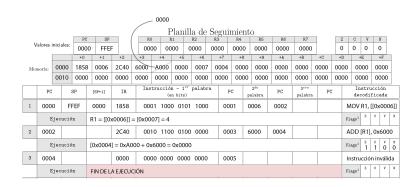
Valor	es l	PC	SP		R0	R	L	R2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les (0000	FFEF]	000	0 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6	000	0000	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	0000	C	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	×0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0×000	4]=0×A00	0+0×600	00=0×0000						Flags 1 1 0 0
3	0004			0000	0000 0000 0000 0000	0005					Instrucción Inválida
	Ejecución										Flags

Valor	es F	C	SP	RO	R	1 1	R2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les (0000	FFEF	00	00 00	000	0000	0000	0000	0000	0000	000	00	0	0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	1858	0006	2C40	6000	0000	0000	0007	0004	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción decodificada
					-		palabra		palabra		
1	0000	FFEF	0000	1858	0001 1000 0101 1000	0001	0006	0002			MOV R1, [[0x0006]]
	Ejecución	R1=[[0	x0006]]=[0×0007]	=4						Flags
2	0002			2C40	0010 1100 0100 0000	0003	6000	0004			ADD [R1],0x6000
	Ejecución	[0×0004	4]=0×A00	$0+0\times600$	00=0×0000						Flags 1 1 0 0
3	0004			0000	0000 0000 0000 0000	0005					Instrucción Inválida
	Ejecución	FIN DE	E LA EJEC	CUCIÓN		Flags					

Ejercicio 1 - Solución



ramación **Ejercicio 1** Ejercicio 2 Ejercicio 3 Cierr ○○○**○○○** ○○○○ ○○○○

Ejercicio 1 - Seguimiento

Ejercicio 1 - Observaciones

¿Qué pasó con el programa anterior?

ramación **Ejercicio 1** Ejercicio 2 Ejercicio 3 Cierr

Ejercicio 1 - Seguimiento

Ejercicio 1 - Observaciones

¿Qué pasó con el programa anterior?

En la ejecución del programa cambió una posición de memoria donde estaba cargado... **el mismo programa!**

ramación **Ejercicio 1** Ejercicio 2 Ejercicio 3 Cierr

Ejercicio 1 - Seguimiento

Ejercicio 1 - Observaciones

¿Qué pasó con el programa anterior?

En la ejecución del programa cambió una posición de memoria donde estaba cargado... **el mismo programa!**

Eso hizo que nuestro programa se modificara, haciendo que la máquina llegase a una instrucción inválida.

ramación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierro

Ejercicio 1 - Seguimiento

Ejercicio 1 - Observaciones

¿Qué pasó con el programa anterior?

En la ejecución del programa cambió una posición de memoria donde estaba cargado... **el mismo programa!**

Eso hizo que nuestro programa se modificara, haciendo que la máquina llegase a una instrucción inválida.

Moraleja:



Ejercicio 2

Realizar el seguimiento de la ejecución del siguiente programa cargado en memoria a partir de la dirección 0x0000.

0000	0001	0002	0003	0004	0005	0006	0007
19C8	0004	B000	0005	0FE0	39A7	C000	0000

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les (0000	FFEF		0000	00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2	-	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D)	+E	+F
0000	19CB	000	4 B000	0	005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	0	0000	0000
0010	0000	000	0000	0	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1											
	Ejecución										Flags
2											
	Ejecución										Flags
3											
	Ejecución										Flags
4											
	Ejecución										Flags
5											
	Ejecución					•					Flags

Ejercicio 2 - Seguimiento

Valor	es F	PC	SP	R0	R1	. F	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 🤇	0000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC		stru codi		
1	0000													╗
	Ejecución				•						Flags			\neg
2														
	Ejecución										Flags			
3														
	Ejecución										Flags			
4														
	Ejecución										Flags			
5														
	Ejecución										Flags			

Ejercicio 2 - Seguimiento

Valor	es	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les 🛭	0000	FFEF		000	0 00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	000	4 B00	0 (0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	000	0000) (0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instr deco	ucció dificad	
1	0000	FFEF	0000										
	Ejecución				•						Flags		
2													
	Ejecución				•						Flags		
3													
	Ejecución										Flags		
4													
	Ejecución										Flags		
5													
	Ejecución			-							Flags		

Ejercicio 2 - Seguimiento

Valor	es	PC	SI	Р	R0	F	R1	R2	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les [0000	FI	FEF	00	0 00	000	00	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	l	+2	+3	+4	+	5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	19CE	3 000	4	B000	0005	0FE0	39/	47	C000	0000	0000	0000	0000	0000	0000	00	00	0000	000
0010	0000	000	0	0000	0000	0000	000	00	0000	0000	0000	0000	0000	0000	0000	00	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada	
1	0000	FFEF	0000	19CB								
	Ejecución										Flags	\Box
2												
	Ejecución				•						Flags	
3												
	Ejecución										Flags	
4												
	Ejecución										Flags	
5												
	Ejecución										Flags	

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les [0000	FFEF		000	0 00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	0004	4 B000	(0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	000
0010	0000	0000	0000	(0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instru decodi	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000		paidbid		parabra		accoun	 \dashv
	Ejecución							•			Flags	П
2												\neg
	Ejecución				•						Flags	
3												
	Ejecución										Flags	
4												
	Ejecución										Flags	
5												
	Ejecución										Flags	

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les [0000	FFEF		000	0 00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	0004	4 B000	(0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	000
0010	0000	0000	0000	(0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC		rucci odific	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001							
	Ejecución				•						Flags		
2													
	Ejecución										Flags		
3													
	Ejecución										Flags		
4													
	Ejecución										Flags		
5													
	Ejecución										Flags		

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z (. V	N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	In	stru	cciói	n	
	r C	31	[31 +1]	1111	Instruccion - 1er parabra	r C	palabra	r C	palabra	FC	de	codi	ficad	la	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002							٦
	Ejecución										Flags			\neg	٦
2															٦
	Ejecución										Flags				
3															
	Ejecución										Flags			\neg	
4															
	Ejecución										Flags			\neg	
5															٦
	Ejecución										Flags		П		
											0-	ш		_	_

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z	V	N
inicia	les 0	000	FFEF	000	00 00	00 00	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Insti			
		J.	[51 1]	"'`	mstrucción ici palabra	' ~	palabra	' ~	palabra	' '	deco	difica	ida	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7	[0x	0004	1
	Ejecución										Flags			
2														
	Ejecución										Flags			
3														
	Ejecución										Flags			
4														
	Ejecución										Flags			
5											·			
	Ejecución										Flags			

Ejercicio 2 - Seguimiento

Valor	es	PC	S	iP	F	R0	R1		R2	R3	R4	R5	R6	R7		Z	С	V	N
inicia	les [0000	F	FEF		0000	00	00 (0000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+	1	+2	+:	3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+	D	+E	+F
0000	19C	B 000)4	B000	000	05 0	FE0	39A7	C00	0 0000	0000	0000	0000	0000	0000	00	00	0000	0000
0010	0000	000	00	0000	000	0 0	000	0000	000	0000	0000	0000	0000	0000	0000	00	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción decodificada
					·		palabra		palabra		
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	<0004]=0>	oFE0							Flags
2											
	Ejecución										Flags
3											
	Ejecución										Flags
4											
	Ejecución										Flags
5											
	Ejecución										Flags

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Ζ	С	V	N
inicia	les (0000	FFEF		0000	00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2	-	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D)	+E	+F
0000	19CB	000	4 B000	0	005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	0	0000	0000
0010	0000	000	0000	0	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Inst	rucci	ión	
	1.0))	[51 +1]	"`	Ilistruccion - 1er parabra	' C	palabra	' '	palabra	' C	deco			
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7	, [0x	:0004	1]
	Ejecución	R7=[0:	×0004]=0×	oFE0							Flags		Т	
2	0002													
	Ejecución										Flags			
3														
	Ejecución										Flags			
4														
	Ejecución										Flags			
5														
	Ejecución										Flags		Т	

Ejercicio 2 - Seguimiento

Valor	es F	PC	SP	R0	R1	. F	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 (0000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Inst				
	1.0	31	[31 +1]	"``	Ilistruccion - 1er parabra	' C	palabra	' '	palabra	' C	deco				
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7	, [0	x00	04]	7
	Ejecución	R7=[0:	<0004]=0>	oFE0							Flags				٦
2	0002			B000											٦
	Ejecución										Flags				7
3															٦
	Ejecución										Flags				
4															7
	Ejecución										Flags				٦
5											, and				٦
	Ejecución										Flags				

Ejercicio 2 - Seguimiento

Valor	es F	PC	SP	R0	R1	. F	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 (0000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Inst	rucci	ón	
	r C	31	[21 +1]	"	mstruccion - 1er palabra	r C	palabra	1	palabra	FC	deco	dific	ada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7	, [0×	0004]	
	Ejecución	R7=[0:	×0004]=0×	oFE0							Flags			
2	0002			B000	1011 0000 0000 0000									
	Ejecución										Flags			
3														
	Ejecución										Flags			
4											·			
	Ejecución										Flags			
5											·			
	Ejecución				·						Flags			

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z	С	V	Ν
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+)	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	0	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0	0000	0000

PC			l IR I	Instrucción - 1er palabra	PC	2da	PC	3ra	PC.	Instr	uccio	on	
	SP	[SP+1]	1111	mstruccion - 1er palabra	r C	palabra	FC	palabra	FC	decod	lifica	da	
0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7,	[0x0	0004	
Ejecución	R7=[0:	×0004]=0×	oFE0							Flags			П
0002			B000	1011 0000 0000 0000	0003								
Ejecución										Flags			
Ejecución										Flags			
Ejecución										Flags			
Ejecución										Flags			
	Ejecución 0002 Ejecución Ejecución	Ejecución R7=[0: 0002 Ejecución Ejecución	Ejecución R7=[0x0004]=0x0002 Ejecución Ejecución Ejecución	Ejecución R7=[0x0004]=0x0FE0 0002 B000 Ejecución	Ejecución R7=[0x004]=0x0FE0 0002 B000 1011 0000 0000 0000 Ejecución	Ejecución R7=[0x0004]=0x0FE0 0002 B000 1011 0000 0000 0000 0003 Ejecución Ejecución Ejecución	Ejecución R7=[0x004]=0x0FE0 0002 B000 1011 0000 0000 0000 0003 Ejecución Ejecuci	Ejecución R7=[0x004]=0x0FE0 0002 B000 1011 0000 0000 0000 0003 Ejecución Ejecuci	Ejecución R7=[0x0004]=0x0FE0 0002 B000 1011 0000 0000 0000 0003 Ejecución Ejecución Ejecución Ejecución	Ejecución R7=[0x004]=0x0FE0 0002 B000 1011 0000 0000 0000 0003 Ejecución Ejecución Ejecución Ejecución	Ejecución R7=[0x0004]=0x0FE0 Flags	Ejecución R7=[0x0004]=0x0FE0	Ejecución R7=[0x0004]=0x0FE0 Flags

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z	CV	N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 (0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	000	0 000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0 000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC.	Inst	ruc	cció	n	
	1.0))	[31 +1]	"``	Ilistruccion - Ter parabra	' C	palabra	' '	palabra	1 C	deco	odit	ficac	la	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R	7, [0x0	004]	П
	Ejecución	R7=[0:	<0004]=0>	oFE0							Flags				
2	0002			B000	1011 0000 0000 0000	0003	0005	0004							
	Ejecución										Flags				
3															
	Ejecución										Flags				
4															
	Ejecución										Flags				
5															
	Ejecución										Flags	T			

Ejercicio 2 - Seguimiento

Valor	es F	PC	SP	R0	R1	. F	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 (0000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Inst	rucci	ón	
	r C	31	[21 +1]	1111	ilistruccion - 1er palabra	r C	palabra	1	palabra	FC	deco	dific	ada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R	', [0×	0004]	٦
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags			٦
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0>	:0005	,	٦
	Ejecución										Flags			
3														٦
	Ejecución										Flags			
4														٦
	Ejecución										Flags			
5														٦
	Ejecución										Flags			

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Inst	rucc	ión	
	FC	31	[21 +1]	1111	mstruccion - 1er palabra	r C	palabra	1	palabra	FC	deco	dific	ada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R	', [0:	<0004	-
	Ejecución	R7=[0:	(0004]=0x	0FE0							Flags			
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0>	:000	5	
	Ejecución	[0×FFE	$F] = 0 \times 000$	4 SP=0	xFFEE PC=0x0005						Flags			
3														\neg
	Ejecución										Flags			
4														
	Ejecución										Flags			
5														
	Ejecución										Flags			

Ejercicio 2 - Seguimiento

Valor	es F	C S	SP	R0	R1	. R	2 F	R3	R4	R5	R6	R7		Z (. V	N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
		0000							0000	0000	0000	0000		0000	0000	

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucc	ión
	FC	35	[21 +1]	"	Instruccion - 1er parabra	1	palabra	r C	palabra	FC	decodific	ada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0:	<0004]
	Ejecución	R7=[0:	×0004]=0×	oFE0	•						Flags	
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x000	5
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags	
3	0005											
	Ejecución										Flags	
4												
	Ejecución										Flags	
5												
	Ejecución										Flags	
5												\pm

Ejercicio 2 - Seguimiento

Valor	es F	PC	SP	R0	R1	. F	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 (0000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción
	FC	31	[21 +1]	"	Instruccion - 1er parabra	1	palabra	r C	palabra	FC	decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0×	oFE0	•						Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	xFFEE PC=0x0005						Flags
3	0005	FFEE									
	Ejecución										Flags
4											
	Ejecución				•						Flags
5											
	Ejecución										Flags
	Ejecución		ı		I	ı					Flags

Ejercicio 2 - Seguimiento

Valor	es F	PC	SP	R0	R1	. F	2	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 (0000	FFEF	000	00 00	00 0	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción	
	r C	31	[21 +1]	"	Instruccion - 1er parabra	1	palabra	r C	palabra	FC	decodificada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]	
	Ejecución	R7=[0:	×0004]=0×	oFE0	•						Flags	
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005	
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags	
3	0005	FFEE	0004	39A7								
	Ejecución										Flags	
4												
	Ejecución										Flags	
5												
	Ejecución										Flags	
		l									8-	-

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z	V	N
inicia	les 0	000	FFEF	000	00 00	00 00	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción
	r C	31	[21 +1]	1111	mstruccion - 1er palabra	r C	palabra	r C	palabra	FC	decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111						
	Ejecución										Flags
4											
	Ejecución										Flags
5											
	Ejecución										Flags
											-

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Z	V	N
inicia	les 0	000	FFEF	000	00 00	00 00	000	0000	0000	0000	0000	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC.	Instrucción	
	FC	31	[21 +1]	1111	mstruccion - 1er palabra	r C	palabra	r C	palabra	FC	decodificada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]	٦
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags	\neg
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005	П
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags	П
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006						П
	Ejecución										Flags	
4												٦
	Ejecución										Flags	П
5												٦
	Ejecución										Flags	\neg
		-										-

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les [0000	FFEF		000	0 00	00 0	000	0000	0000	0000	0000	000	0	0	0	0	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	0004	4 B000	(0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	000
0010	0000	0000	0000	(0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da	PC	3ra	PC	Instrucción
	r C	31	[21 +1]	1111	ilistruccion - 1er palabra	r C	palabra	r C	palabra	FC	decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución										Flags
4											
	Ejecución										Flags
5											
	Ejecución										Flags
											. 6.

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	F	R2	R3	R4	R5	R6	R7		Z	١ ١	/	N
inicia	les [0000	FFEF		000	0 00	00 (0000	0000	0000	0000	0000	000	00	0 () ()	0
	+0	+1	+2		+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+	E	+F
0000	19CB	0004	4 B000	0	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	00	000	0000
0010	0000	0000	0000	0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	00	00	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
									palabra		
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags
4											
	Ejecución										Flags
5											
	Ejecución										Flags

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. F	22	R3	R4	R5	R6	R7		Z	. V	N
inicia	les 🛚 🤇	0000	FFEF	00	00 00	00 0	000	0000	0000	0000	F020	000	00	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002	paiabia		MOV R7, [0x0004]
_	Ejecución	R7=[0:	<0004]=0>	oFE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
	Ejecución	[0xFFE	F]=0x000	4 SP=0:	xFFEE PC=0x0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4											
	Ejecución										Flags
5											
	Ejecución										Flags

Ejercicio 2 - Seguimiento

Valor	es F	C !	SP	R0	R1	R	2 F	R3	R4	R5	R6	R7		Z		N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	F020	000	0	0 (0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC.	2da	PC.	3ra	PC.	Instrucción	٦
	r C	31	[21 +1]	1111	ilistruccion - 1er palabra	r C	palabra	r C	palabra	FC	decodificada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]	٦
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags	٦
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005	٦
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags	٦
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7	٦
	Ejecución	R6=0x	:0000-0×0F	E0=0×F	020						Flags 0 1 0 1	П
4	0006											٦
	Ejecución										Flags	٦
5												٦
	Ejecución										Flags	٦
											. 0.	_

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les [0000	FFEF		000	00	00 0	000	0000	0000	0000	F020	000	0	0	0	0	0
	+0	+1	+2	2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	000	4 B00	00 (0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	000
0010	0000	000	000	0 (0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC.	2da	PC.	3ra	PC.	Instrucción	
	FC	31	[21 +1]	IIX	ilistruccion - 1er palabra	r C	palabra	r C	palabra	FC	decodificada	
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x000)4]
	Ejecución	R7=[0:	×0004]=0×	0FE0							Flags	
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005	
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags	
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7	
	Ejecución	R6=0x	:0000-0×0F	E0=0×F	020						Flags 0 1 (1
4	0006			C000								
	Ejecución										Flags	
5												
	Ejecución										Flags	
											0-	_

Ejercicio 2 - Seguimiento

Valor	es F	C !	SP	R0	R1	R	2 F	R3	R4	R5	R6	R7		Z		N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	F020	000	0	0 (0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC.	2da	PC.	3ra	PC	Instrucción
	r C	31	[21 +1]	"	mstruccion - 1er palabra	1	palabra	r C	palabra	FC	decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0×	oFE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0×FFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	:0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000						
	Ejecución										Flags
5											
	Ejecución										Flags
											. 0.

Ejercicio 2 - Seguimiento

Valor	es F	C !	SP	R0	R1	R	2 F	R3	R4	R5	R6	R7		Z	CV	N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	F020	000	0	0 (0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000

PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC.	2da	PC.	3ra	PC	Instrucción	
rc	3F	[21 +1]	IIX	mstruccion - 1er palabra	r C	palabra	r C	palabra	FC	decodificada	а
0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x00	04]
Ejecución	R7=[0	×0004]=0×	0FE0							Flags	
0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005	
Ejecución	[0xFFE	$F] = 0 \times 000$	4 SP=0:	×FFEE PC=0×0005						Flags	
0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7	
Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1	0 1
0006			C000	1100 0000 0000 0000	0007						
Ejecución										Flags	
Ejecución										Flags	
	0000 Ejecución 0002 Ejecución 0005 Ejecución 0006 Ejecución	0000 FFEF Ejecución R7=[0: 0002 Ejecución [0xFFE 0005 FFEE Ejecución R6=0x 0006 Ejecución	0000 FFEF 0000 Ejecución R7=[0x0004]=0x 0002 Ejecución [0xFFEF]=0x000 0005 FFEE 0004 Ejecución R6=0x0000-0x0F 0006 Ejecución	0000 FFEF 0000 19CB Ejecución R7=[0x0004]=0x0FE0 0002 B000 Ejecución [0xFFEF]=0x0004 SP=0; 0005 FFEE 0004 39A7 Ejecución R6=0x0000-0x0FE0=0xF 0006 C000 Ejecución Ejecución C000	0000 FFEF 0000 19CB 0001 1001 1100 1000 Ejecución R7=[0x0004]=0x0FE0 0002 B000 1011 0000 0000 0000 Ejecución [0xFFEF]=0x0004 SP=0xFFEE PC=0x0005 0005 FFEE 0004 39A7 0011 1001 1010 0111 Ejecución R6=0x0000-0x0FE0=0xF020 0006 C000 1100 0000 0000 0000 Ejecución C000 1100 0000 0000 0000	0000 FFEF 0000 19CB 0001 1001 1100 1000 0001 Ejecución R7=[0x0004]=0x0FE0 0002 B000 1011 0000 0000 0000 0003 Ejecución [0xFFEF]=0x0004 SP=0xFFEE PC=0x0005 0005 FFEE 0004 39A7 0011 1001 1010 0111 0006 Ejecución R6=0x000-0x0FE0=0xF020 0006 0006 0007 0007 Ejecución Ejecución 0000 0007 0007 0000	D000	0000	Description Description		Moverage Moverage

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les [0000	FFEF		000	00	00 0	000	0000	0000	0000	F020	000	0	0	0	0	0
	+0	+1	+2	2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	000	4 B00	00 (0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	000
0010	0000	000	000	0 (0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC.	2da	PC.	3ra	PC	Instrucción
	r C	31	[31 +1]	"	mstruccion - 1er palabra	1	palabra	r C	palabra	FC	decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	<0004]=0>	oFE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0×FFE	$F] = 0 \times 000$	4 SP=0:	×FFEE PC=0×0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución										Flags
5											
	Ejecución										Flags

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP		R0	R1	R	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les [0000	FFEF		000	00	00 0	000	0000	0000	0000	F020	000	0	0	0	0	0
	+0	+1	+2	2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CE	000	4 B00	00 (0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	000
0010	0000	000	000	0 (0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002	parabra		MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0>	oFE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
	Ejecución	[0xFFE	F]=0x000	4 SP=0:	FFEE PC=0x0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0xF	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución	PC=[0	xFFEF]=0)×0004 S	P=0xFFEF						Flags
5											
	Ejecución										Flags

amación Ejercicio 1 **Ejercicio 2** Ejercicio 3 Cierre 000000 0**00** 00000

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP	F	R0	R1	R	2	R3	R4	R5	R6	R7		Z		N	
inicia	les (0000	FFEF		0000	000	00 00	000	0000	0000	0000	F020	000	0	0 (0	0]
	+0	+1	+2	+3	3 -	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+1	<u> </u>	+F
0000	19CB	0004	1 B000	000)5 OF	FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	000	0 0	000
0010	0000	0000	0000	000	00 00	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0 0	000

0000				Instrucción - 1er palabra	PC	palabra	PC	palabra	PC	Instrucción decodificada
	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
Ejecución	R7=[0:	×0004]=0×	0FE0							Flags
0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
Ejecución	[0xFFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags
0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
0006			C000	1100 0000 0000 0000	0007					RET
Ejecución	PC=[0:	xFFEF]=0	×0004 S	P=0xFFEF						Flags
0004										
Ejecución										Flags
E OI	jecución 005 jecución 006 jecución 004	002 jecución [0xFFE 005 FFEE jecución R6=0x 006 jecución PC=[0	002 jecución [0xFFEF]=0x000 005 FFEE 0004 jecución R6=0x0000-0x0F 006 jecución PC=[0xFFEF]=C 004	B000 jecución [0xFFEF]=0x0004 SP=0: 005 FFEE 0004 39A7 jecución R6=0x0000-0x0FE0=0xF 006 C000 jecución PC=[0xFFEF]=0x0004 S	B000 1011 0000 0000 0000		B000 1011 0000 0000 0000 0003 0005	B000 1011 0000 0000 0000 0003 0005 0004		

mación Ejercicio 1 **Ejercicio 2** Ejercicio 3 Cierre 0000000 0**000** 00000

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	. R	2	R3	R4	R5	R6	R7		Ζ	C	V	N
inicia	les 0	000	FFEF	000	00 00	00 0	000	0000	0000	0000	F020	000	0	0	0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+[)	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	00	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	00	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0>	0FE0		•					Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
	Ejecución	[0xFFE	F]=0x000	4 SP=0:	×FFEE PC=0×0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución	PC=[0	xFFEF]=0	×0004 S	P=0xFFEF						Flags
5	0004	FFEF	0000								
	Ejecución				•						Flags
5	0004			1×0004 S	P=0xFFEF						

Ejercicio 2 - Seguimiento

Valor	es F	C	SP	R0	R1	R	2	R3	R4	R5	R6	R7		Z	С	V	N
inicia	les 🛚 C	0000	FFEF	000	00 00	00 00	000	0000	0000	0000	F020	000	0	0	0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+)	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	000	0	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002	Fulleand		MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0>	oFE0		•					Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
	Ejecución	[0xFFE	F]=0x000	4 SP=0:	xFFEE PC=0x0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución	PC=[0	xFFEF]=0)×0004 S	P=0xFFEF						Flags
5	0004	FFEF	0000	0FE0							
	Ejecución										Flags

amación Ejercicio 1 **Ejercicio 2** Ejercicio 3 Cierro 000000 0**00** 00000

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP	F	R0	R1	R	2	R3	R4	R5	R6	R7		Z		N	
inicia	les (0000	FFEF		0000	000	00 00	000	0000	0000	0000	F020	000	0	0 (0	0]
	+0	+1	+2	+3	3 -	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+1	<u> </u>	+F
0000	19CB	0004	1 B000	000)5 OF	FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	000	0 0	000
0010	0000	0000	0000	000	00 00	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0 0	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	<0004]=0×	0FE0							Flags
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0xFFE	F]=0x000	4 SP=0	FFEE PC=0x0005						Flags
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución	PC=[0:	xFFEF]=0	×0004 S	P=0×FFEF						Flags
5	0004	FFEF	0000	0FE0	0000 1111 1110 0000						
	Ejecución										Flags

Ejercicio 2 - Seguimiento

Ejercicio 2 - Seguimiento

Valor	es I	PC	SP		R0	R1	F	22	R3	R4	R5	R6	R7		Z	C	V	N
inicia	les [0000	FFE	ΕF	000	00	00 0	000	0000	0000	0000	F020	000	0	0	0	0	0
	+0	+1	+	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	-	+E	+F
0000	19CE	000	4 B(000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0 0	000	000
0010	0000	000	0 00	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000) ()	000	000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002			MOV R7, [0x0004]
	Ejecución	R7=[0:	×0004]=0>	OFE0						Flags	
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0×0005
	Ejecución	[0xFFE	[0xFFEF]=0x0004 SP=0xFFEE PC=0x0005						Flags		
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución	PC=[0	PC=[0xFFEF]=0x0004 SP=0xFFEF Flags						Flags		
5	0004	FFEF	0000	0FE0	0000 1111 1110 0000	0005					
	Ejecución										Flags

Ejercicio 2 - Seguimiento

Ejercicio 2 - Seguimiento

Valor	es l	PC	SP	F	R0	R1	R	2	R3	R4	R5	R6	R7		Z		N	
inicia	les (0000	FFEF		0000	000	00 00	000	0000	0000	0000	F020	000	0	0 (0	0]
	+0	+1	+2	+3	3 -	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+1	<u> </u>	+F
0000	19CB	0004	1 B000	000)5 OF	FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	000	0 0	000
0010	0000	0000	0000	000	00 00	000	0000	0000	0000	0000	0000	0000	0000	0000	0000	000	0 0	000

Instrucción
decodificada
OV R7, [0x0004]
ags
ALL 0×0005
ags
JB R6, R7
ags 0 1 0 1
ĒΤ
ags
strucción Inválida
ags

Ejercicio 2 - Seguimiento

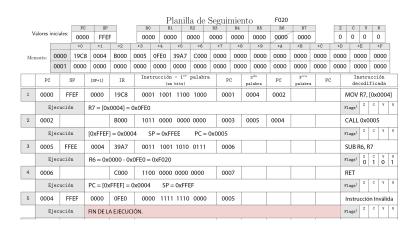
Ejercicio 2 - Seguimiento

Valor	es P	'C :	SP	R0	R1	R	2 1	R3	R4	R5	R6	R7		Z	C V	N
inicia	les 0	000	FFEF	000	00 00	00 00	000	0000	0000	0000	F020	000	0	0 0	0	0
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	19CB	0004	B000	0005	0FE0	39A7	C000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

	PC	SP	[SP+1]	IR	Instrucción - 1er palabra	PC	2da palabra	PC	3ra palabra	PC	Instrucción decodificada
1	0000	FFEF	0000	19CB	0001 1001 1100 1000	0001	0004	0002	paiabia		MOV R7, [0x0004]
	Ejecución	R7=[0x0004]=0x0FE0								Flags	
2	0002			B000	1011 0000 0000 0000	0003	0005	0004			CALL 0x0005
	Ejecución	[0xFFE	0xFFEF]=0x0004 SP=0xFFEE PC=0x0005 Flags						Flags		
3	0005	FFEE	0004	39A7	0011 1001 1010 0111	0006					SUB R6, R7
	Ejecución	R6=0x	0000-0×0F	E0=0×F	020						Flags 0 1 0 1
4	0006			C000	1100 0000 0000 0000	0007					RET
	Ejecución	PC=[0	PC=[0xFFEF]=0x0004 SP=0xFFEF						Flags		
5	0004	FFEF	0000	0FE0	0000 1111 1110 0000	0005					Instrucción Inválida
	Ejecución	FIN DE	FIN DE LA EJECUCIÓN Flags Flags								
	J										. 0

Eiercicio 2 - Seguimiento

Ejercicio 2 - Solución



Ejercicio 2 - Seguimiento

Ejercicio 2 - Observaciones

¿Qué pasó con el programa anterior?

Ejercicio 2 - Observaciones

¿Qué pasó con el programa anterior?

- Utilizamos la instrucción CALL.
- El valor del PC de retorno se almacena en la pila durante de la ejecución de la instrucción.
- Recuerden que la pila crece en el sentido de las direcciones de memoria menores.
- Al regresar con RET, se "saca" la dirección de retorno de allí.

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

> main: MOV R1, 0x0000

loop: CMP R1, 0×0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: **RET**

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RET

dirección ocupa

0×0000

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

0x0000 dos palabras

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

0×0002

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

 $\sf JL\ sumoUno$

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

 0×0002 dos palabras

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

0x0002 dos palabras

0×0004

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

 0×0002 dos palabras

0x0004 una palabra

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

 0×0002 dos palabras

0x0004 una palabra

0×0005

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

0x0002 dos palabras

0x0004 una palabra

0x0005 una palabra

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

 0×0002 dos palabras

0x0004 una palabra

0x0005 una palabra

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JL sumoUno

JGE fin

sumoUno: ADD R1, 0x0001

JNE loop

fin: RFT

dirección ocupa

 0×0000 dos palabras

0x0002 dos palabras

0x0004 una palabra

0x0005 una palabra

 0×0006 dos palabras

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R	21, 0×0000
-------------	------------

loop: CMP R1, 0x0001

JL sumoUno

sumoUno: ADD R1, 0x0001

JNE loop

JGE fin

fin: RET

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	una palabra
0×0005	una palabra
0×0006	dos palabras
0×0008	

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main: MOV R1, 0x0000

loop: CMP R1, 0x0001

JGE fin

JL sumoUno

ADD R1, 0×0001

JNE loop

fin: RFT

dirección ocupa

0x0000 dos palabras

0x0002 dos palabras

0x0004 una palabra

0x0005 una palabra

0x0006 dos palabras

una palabra

8000x0

sumoUno:

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main:	MOV R1, 0×0000
loop:	CMP R1, 0x0001
	JL sumoUno
	JGE fin
sumoUno:	ADD R1, 0x0001
	JNE loop
fin:	RET

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	una palabra
0×0005	una palabra
0×0006	dos palabras
8000×0	una palabra
0×0009	

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

main:	MOV R1, 0×0000
loop:	CMP R1, 0x0001
	JL sumoUno
	JGE fin
sumoUno:	ADD R1, 0x0001
	JNE loop
fin:	RET

dirección	ocupa
0×0000	dos palabras
0×0002	dos palabras
0×0004	una palabra
0×0005	una palabra
0×0006	dos palabras
0×0008	una palabra
0×0009	una palabra

Codifiquemos el siguiente código y carguémoslo desde la posición de memoria 0x0000:

		dirección	ocupa
main:	MOV R1, 0×0000	0×0000	dos palabras
loop:	CMP R1, 0x0001	0×0002	dos palabras
	JL sumoUno	0×0004	una palabra
	JGE fin	0×0005	una palabra
sumoUno:	ADD R1, 0x0001	0×0006	dos palabras
	JNE loop	0×0008	una palabra
fin:	RET	0×0009	una palabra

Buenísimo, tenemos las direcciones de las etiquetas, pero... *los saltos condicionales son* **relativos**.

Ejercicio 3 - Ensamblado

Buenísimo, tenemos las direcciones de las etiquetas, pero... los saltos condicionales son relativos.

• Esto significa que las etiquetas de los saltos son reemplazadas por el desplazamiento necesario para "llegar" desde la dirección en la que estamos parados, hasta la de la etiqueta de destino.

Ejercicio 3 - Ensamblado

Buenísimo, tenemos las direcciones de las etiquetas, pero... los saltos condicionales son relativos.

- Esto significa que las etiquetas de los saltos son reemplazadas por el desplazamiento necesario para "llegar" desde la dirección en la que estamos parados, hasta la de la etiqueta de destino.
- Vamos a tener que calcular estos desplazamientos.

0×0002: CMP R1, 0×0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \mathsf{_Etiqueta} - Dir \mathsf{_Instr}_{postfetch}$

0x0000: MOV R1, 0x0000

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} _\mathsf{Etiqueta} - Dir _\mathit{Instr}_{postfetch}$

desplazamiento sumoUno:

0x0000: MOV R1, 0x0000

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir_Etiqueta} - Dir_Instr_{postfetch}$

desplazamiento sumoUno:

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \mathsf{_Etiqueta} - Dir \mathsf{_Instr}_{postfetch}$

desplazamiento sumoUno:

0×0006 - 0×0005

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno:

 $0 \times 0006 - 0 \times 0005 = 1$

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} _ \mathsf{Etiqueta} - Dir _ Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} _\mathsf{Etiqueta} - Dir _\mathit{Instr}_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin: $0 \times 0009 - 0 \times 0006$

0x0000: MOV R1, 0x0000

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin: $0 \times 0009 - 0 \times 0006 = 3$

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno:

 $0 \times 00006 - 0 \times 00005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

desplazamiento loop:

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

desplazamiento loop:

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

desplazamiento loop:

 $0 \times 0002 - 0 \times 0009$

Ejercicio 3 - Ensamblado

0x0000: MOV R1, 0x0000

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} - Dir \ldotp \mathit{Instr}_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

desplazamiento loop: $0 \times 0002 - 0 \times 0009 = -7$

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} \cdot Dir \ldotp Instr_{postfetch}$

desplazamiento sumoUno:

 $0 \times 00006 - 0 \times 00005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

desplazamiento loop:

 $0 \times 0002 - 0 \times 0009 = -7 = 0 \times F9$

0x0002: CMP R1, 0x0001

0x0004: JL sumoUno

0x0005: JGE fin

0×0006: ADD R1, 0×0001

0x0008: JNE loop

0x0009: RET

 $\mathsf{Despl} = \mathsf{Dir} \ldotp \mathsf{Etiqueta} - Dir \ldotp \mathit{Instr}_{postfetch}$

desplazamiento sumoUno: $0 \times 0006 - 0 \times 0005 = 1 = 0 \times 01$

desplazamiento fin:

 $0 \times 0009 - 0 \times 0006 = 3 = 0 \times 03$

desplazamiento loop:

 $0 \times 0002 - 0 \times 0009 = -7 = 0 \times F9$

0x0002: CMP R1, 0x0001

0x0004: JL 0x01

0x0005: JGE 0x03

0x0006: ADD R1, 0x0001

0x0008: JNE 0xF9

0x0009: RET

Ahora sí: nuestro programa esta listo para ser ensamblado!

Tarea: ensamblado y segui-

miento!

Ejercicio 3 - Observaciones

Ejercicio 3 - Ensamblado

Ejercicio 3 - Ensamblado

Ejercicio 3 - Observaciones

Moraleja del ejercicio anterior:

 Los desplazamientos de los saltos condicionales siempre son relativos.

Ejercicio 3 - Observaciones

- Los desplazamientos de los saltos condicionales siempre son relativos.
- Siempre van a ser traducidas las etiquetas primero, y luego calculados los desplazamientos.

Ejercicio 3 - Observaciones

- Los desplazamientos de los saltos condicionales siempre son relativos.
- Siempre van a ser traducidas las etiquetas primero, y luego calculados los desplazamientos.
- Para calcular los desplazamientos, tengan en cuenta que el salto se ejecuta *después* de incrementar el PC.

- Los desplazamientos de los saltos condicionales siempre son relativos.
- Siempre van a ser traducidas las etiquetas primero, y luego calculados los desplazamientos.
- Para calcular los desplazamientos, tengan en cuenta que el salto se ejecuta después de incrementar el PC.
- Los desplazamientos se representan en complemento a dos.

Ejercicio 3 - Observaciones

- Los desplazamientos de los saltos condicionales siempre son relativos.
- Siempre van a ser traducidas las etiquetas primero, y luego calculados los desplazamientos.
- Para calcular los desplazamientos, tengan en cuenta que el salto se ejecuta después de incrementar el PC.
- Los desplazamientos se representan en complemento a dos.
- Siempre extender los desplazamientos con signo a la hora de aplicarlos.

Resumen

Hoy vimos:

- Máquina Orga 1 y su arquitectura
 - ISA: formato de instrucción, modos de direccionamiento
 - ciclo de ejecución
- Ciclo de vida de un programa
 - uso de etiquetas
 - directivas
- Ensamblamos programas
- Seguimos programas dentro de la máquina Orga1

¿Preguntas?

¿Preguntas?



Programación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierre

Avisos

Martes 8/5: Taller de Ciclo de Instrucción en los labos. IMPORTANTE: Traer Cartilla Orga 1 y planillas de Seguimiento

Programación Ejercicio 1 Ejercicio 2 Ejercicio 3 Cierre

¡Eso es todo amigos!

