

Arreglos

Organización de computadoras

Universidad Nacional de Quilmes

June 16, 2020

Arreglos

Arreglo

Arreglo de valores

- Colección de elementos, contenidos en

Arreglo

Arreglo de valores

- Colección de elementos, contenidos en
- posiciones de memoria consecutivas, donde

⋮	⋮
000A	1er valor
000B	2do valor
000C	3er valor
000D	4to valor
000E	5to valor
000F	6to valor
⋮	⋮

Arreglo

Arreglo de valores

- Colección de elementos, contenidos en
- posiciones de memoria consecutivas, donde
- cada elemento puede ocupar mas de una celda, o menos.

⋮	⋮
000A	1er valor
000B	2do valor
000C	3er valor
000D	4to valor
000E	5to valor
000F	6to valor
⋮	⋮

Arreglo

Arreglo de valores

- Colección de elementos, contenidos en
- posiciones de memoria consecutivas, donde
- cada elemento puede ocupar mas de una celda, o menos.
- Pero todos ocupan lo mismo

⋮	⋮
000A	1er valor
000B	2do valor
000C	3er valor
000D	4to valor
000E	5to valor
000F	6to valor
⋮	⋮

Arreglos


Tamaño de un arreglo
Cantidad de elementos del arreglo

Arreglos


Tamaño de un arreglo

Cantidad de elementos del arreglo

⋮	⋮
000A	1110
000B	2345
000C	A0A9
000D	FEDC
000E	0000
000F	FFFF
⋮	⋮

Un elemento por celda  tamaño=6

⋮	⋮
000A	1110
000B	2345
000C	A0A9
000D	FEDC
000E	0000
000F	FFFF
⋮	⋮

Cada elemento ocupa 2 celdas  tamaño=3

Tamaño del arreglo

¿Dónde termina un arreglo?

Tamaño del arreglo

¿Dónde termina un arreglo?



Se conoce la cantidad de elementos (tamaño)

O

Se conoce una condición de fin

Desafío

Desafío

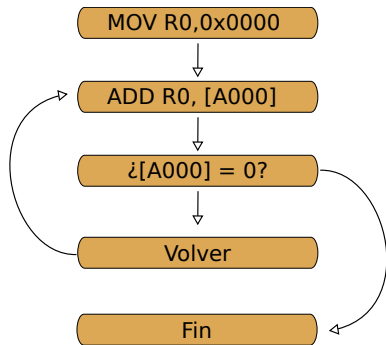


A partir de la celda A000 hay un arreglo que contiene los pedidos de empanadas de una rotisería, y que finaliza con el primer valor 0. Sumar todos los valores

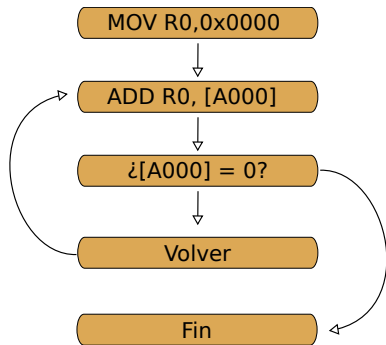
⋮	⋮
A000	0010
A001	001A
A002	0014
A003	0018
A004	0000
⋮	⋮

A partir de la celda A000 hay un arreglo que contiene los pedidos de empanadas de una rotisería, y que finaliza con el primer valor 0. Sumar todos los valores

A partir de la celda A000 hay un arreglo que contiene los pedidos de empanadas de una rotisería, y que finaliza con el primer valor 0. Sumar todos los valores



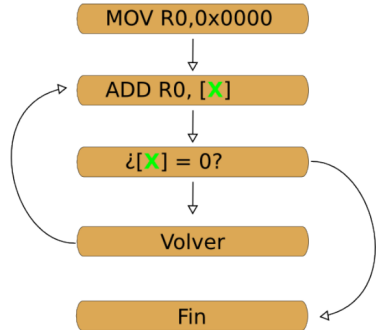
A partir de la celda A000 hay un arreglo que contiene los pedidos de empanadas de una rotisería, y que finaliza con el primer valor 0. Sumar todos los valores



¿Que limitación encontramos?

¡La dirección del operando
debe variar!

¡La dirección del operando
debe variar!



Lo que estabas necesitando
es...

Lo que estabas necesitando
es...



¡Modos de direccionamiento
indirecto!

Modos indirectos

Modo de direccionamiento indirecto por registro

La dirección de memoria del operando está en un registro

Modos indirectos

Modo de direccionamiento indirecto por registro

La dirección de memoria del operando está en un registro



```
MOV R0, [R5]
```

Modos indirectos

Modo de direccionamiento indirecto por registro

La dirección de memoria del operando está en un registro



```
MOV R0, [R5]
```



¿Dónde está el operando?

Modos indirectos

MOV R0, [R5]

¿Dónde está el operando?

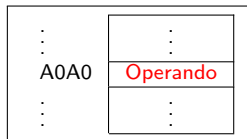
Modos indirectos

MOV R0, [R5]

¿Dónde está el operando?



R5 = A0A0



Indirecto por registro: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

```
ADD R0, [R5]
```

Indirecto por registro: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

ADD R0, [R5]



1 lectura

Indirecto por registro: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

ADD R0, [R5]



1 lectura

¿Cuántos accesos a memoria ocurren?

ADD [R0], R5

Indirecto por registro: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

ADD R0, [R5]



1 lectura

¿Cuántos accesos a memoria ocurren?

ADD [R0], R5



1 acceso de lectura + 1 escritura

Modos indirectos

Modo de direccionamiento indirecto

La **dirección de memoria** del operando está en una celda

Modos indirectos

Modo de direccionamiento indirecto

La dirección de memoria del operando está en una celda



```
MOV R0, [[FFFF]]
```

Modos indirectos

Modo de direccionamiento indirecto

La dirección de memoria del operando está en una celda



```
MOV R0, [[FFFF]]
```



¿Dónde está el operando?

Modos indirectos

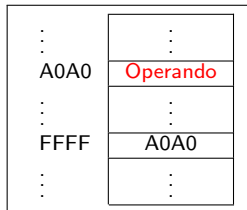
```
MOV R0, [[FFFF]]
```

¿Dónde está el operando?

Modos indirectos

MOV R0, [[FFFF]]

¿Dónde está el operando?



Indirecto: accesos a Memoria

¿Cuántos accesos a memoria ocurren?
ADD R0, [[FAFA]]

Indirecto: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

ADD R0, [[FAFA]]



2 lecturas

Indirecto: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

```
ADD R0, [[FAFA]]
```



2 lecturas

¿Cuántos accesos a memoria ocurren?

```
ADD [[FAFA]], R5
```

Indirecto: accesos a Memoria

¿Cuántos accesos a memoria ocurren?

```
ADD R0, [[FAFA]]
```



2 lecturas

¿Cuántos accesos a memoria ocurren?

```
ADD [[FAFA]], R5
```



2 lecturas + 1 escritura (la dirección queda calculada)

Ejercicio

⋮	⋮
0000	0007
0001	0000
0002	2345
0003	F0F0
0004	04F5
0005	0001
⋮	⋮

R1=0005 y R2=0003
¿Que valor queda en R0?

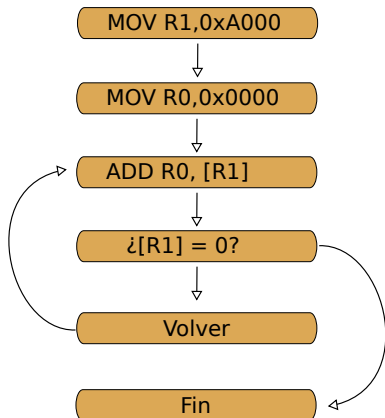
- 1 MOV R0, [[0005]]
- 2 MOV R0, [[0001]]
- 3 MOV R0, [R1]
- 4 MOV R0, [R2]

Revisando el programa de la rotisería

A partir de la celda A000 hay un arreglo que contiene los pedidos de empanadas de una rotisería, y que finaliza con el primer valor 0. Sumar todos los valores

Revisando el programa de la rotisería

A partir de la celda A000 hay un arreglo que contiene los pedidos de empanadas de una rotisería, y que finaliza con el primer valor 0. Sumar todos los valores



Ejercicio: Completar el programa

Ejercicio: Calcular el promedio de un arreglo de temperaturas que comienza en la celda B0B0 y finaliza con el primer 0.

Ejercicio: Calcular el promedio de un arreglo de temperaturas que comienza en la celda B0B0 y finaliza con el primer 0.

```

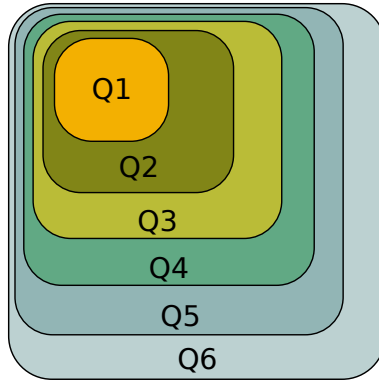
;-----PromedioB0B0
; Requiere: A partir de B0B0 un arreglo de valores que finaliza en un 0
; Retorna: Promedio en R2
;-----
promedioB0B0: MOV R2, 0x0000 ; ACUMULADOR
              MOV R1, 0x0000 ; CONTADOR
              MOV R0, 0xB0B0 ; dirección inicial
volver: CMP [R0], 0x0000 ; ¿está vacío?
        JE fin
        ADD R2, [R0]; acumula el valor actual
        ADD R1, 0x0001; cuento un elemento
        ADD R0, 0x0001; avanzo en el arreglo
        JMP volver ; fin del ciclo
fin:    CMP R1, 0x0000; para no dividir por 0
        JE salir
        DIV R2,R1; calcula el promedio
salir: RET
  
```

Arquitecturas Q

... La que te faltaba ...

Arquitectura Q6

Arquitectura Q6



Arquitectura Q6

- Tiene 8 registros de uso general de 16 bits: R0..R7
- Tiene direcciones de 16 bits
- Los operandos pueden estar en registros, ser constantes o estar en direcciones de memoria
- permite 3 modos de direccionamiento:
 - modo registro: el valor buscado está en un registro
 - modo inmediato: el valor buscado está codificado dentro de la instrucción
 - modo directo: el valor buscado está contenido en una celda de memoria
 - modo indirecto: la dirección del valor buscado está contenido en una celda de memoria
 - modo registro indirecto: la dirección del valor buscado está contenido en un registro

Arquitectura Q6: formato de instrucciones

- Instrucciones de 2 operandos
(MUL,MOV,ADD,SUB,CMP,DIV,AND,OR)

Cod_Op (4b)	Modo Destino (6b)	Modo Origen (6b)	Operando Destino (16b)	Operando Origen (16b)
----------------	----------------------	---------------------	---------------------------	--------------------------

- Instrucciones con un operando Origen: CALL, JMP

Cod_Op (4b)	Relleno (000000)	Modo Origen (6b)	Operando Origen (16b)
----------------	---------------------	---------------------	--------------------------

- Instrucciones con un operando Destino: NOT

Cod_Op (4b)	Modo Destino (6b)	Relleno (000000)	Operando Origen (16b)
----------------	----------------------	---------------------	--------------------------

- Instrucciones sin operandos: RET

Cod_Op (4b)	Relleno (000000000000)
----------------	---------------------------

- Saltos **condicionales** y relativos

Prefijo (1111)	Cod_Op (4)	Desplazamiento(8) (8b)
-------------------	---------------	---------------------------

(Idem Q5)

Arquitectura Q6: Instrucciones Aritméticas

Cod_Op (4b)	Modo Destino (6b)	Modo Origen (6b)	Operando Destino (16b)	Operando Origen (16b)
		Operación	CodOp	
		MUL	0000	
		MOV	0001	
		ADD	0010	
		SUB	0011	
		AND	0100	
		OR	0101	
		CMP	0110	
		DIV	0111	

(Idem Q5)

Arquitectura Q6: Instrucciones sin operandos

Tipo 5: Instrucciones sin operandos

		Cod_Op (4b)	Relleno (000000000000)
Operación	CodOp	Efecto	
RET	1100	$PC \leftarrow [SP+1]; SP \leftarrow SP + 1$	

(Idem Q5)

Arquitectura **Q6**: Instrucciones con un operando Origen

Cod_Op (4b)	Relleno (000000)	Modo Origen (6b)	Operando Origen (16b)
----------------	---------------------	---------------------	--------------------------

Operación	CodOp	Efecto
JMP	1010	PC \leftarrow Origen
CALL	1011	[SP] \leftarrow PC; SP \leftarrow SP - 1; PC \leftarrow Origen

El operando Origen en el JMP es una dirección absoluta

(Idem **Q5**)

Arquitectura Q6: Instrucciones con un operando Destino

Cod_Op (4b)	Modo Destino (6b)	Relleno (000000)	Operando Origen (16b)
----------------	----------------------	---------------------	--------------------------

Operación	CodOp	Efecto
NOT	1001	Dest \leftarrow NOT Dest

(Idem Q5)

Arquitectura Q6: Instrucciones con un operando Destino

Cod_Op (4b)	Modo Destino (6b)	Relleno (000000)	Operando Origen (16b)
----------------	----------------------	---------------------	--------------------------

Operación	CodOp	Efecto
NOT	1001	Dest \leftarrow NOT Dest

(Idem Q5)

Arquitectura Q6: Saltos condicionales (relativo) - 1 de 2

Prefijo (1111)	Cod_Op (4)	Desplazamiento(8) (8b)
-------------------	---------------	---------------------------

En todos los saltos, el efecto es:

$$PC \leftarrow PC + \text{desplazamiento}$$

Salto	Codop	Descripción	Condición
JE	0001	Igual / Cero	Z
JNE	1001	No igual	\overline{Z}
JLEU	0100	Menor o igual sin signo	$C^V Z$
JGU	1100	Mayor sin signo	$\overline{(C^V Z)}$
JCS	0101	Menor sin signo	C
JNEG	0110	Negativo	N

(Idem Q5)

Arquitectura Q6: Saltos condicionales (relativo) - 2 de 2

Prefijo (1111)	Cod_Op (4)	Desplazamiento(8) (8b)
-------------------	---------------	---------------------------

Salto	Codop	Descripción	Condición
JVS	0111	Overflow	V
JLE	0010	Menor o igual con signo	$Z^V(N \oplus V)$
JG	1010	Mayor con signo	$\overline{(Z^V(N \oplus V))}$
JL	0011	Menor con signo	$N \oplus V$
JGE	1011	Mayor o igual con signo	$\overline{(N \oplus V)}$

(Idem Q5)

Arquitectura Q6: Códigos de los modos de direccionamiento

Modo	Codificación
Inmediato	000000
Directo	001000
Indirecto	011000
Registro	100rrr
Registro indirecto	110rrr

donde rrr es una codificación (en 3 bits) del número de registro.

Arquitectura Q6: Ejercicio

Ensamblar el siguiente programa, ubicándolo a partir de la celda 0FF0

```
MOV R1, [[0x0000]]  
volver CMP [R2], 0x0000
```

Arquitectura Q6: Ejercicio

Ensamblar el siguiente programa, ubicándolo a partir de la celda 0FF0

```
MOV R1, [[0x0000]]  
volver CMP [R2], 0x0000
```



completar la tabla de accesos

Instrucción	B.Inst.	B.Op.	Alm.Op.

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 Comparar el valor actual con el mínimo
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 Volver al paso 3

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 Comparar el valor actual con el mínimo
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 Volver al paso 3

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 **Comparar el valor actual con el mínimo**
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 Volver al paso 3

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 Comparar el valor actual con el mínimo
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 Volver al paso 3

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 Comparar el valor actual con el mínimo
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 Volver al paso 3

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 Comparar el valor actual con el mínimo
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 Volver al paso 3

(1) Hacer un programa que encuentre el máximo valor de un arreglo, el arreglo comienza en la celda ABAB y tiene 80 elementos en CA2(16)

Idea!

- 1 Inicializar un registro con el mínimo representable en el sistema CA2(16)
- 2 Inicializar un registro que lleve la dirección de cada elemento
- 3 Comparar el valor actual con el mínimo
- 4 Si el valor es mayor, seguir con el siguiente
- 5 Si es menor, reemplazar el mínimo con el valor actual
- 6 Incrementar el registro de la dirección
- 7 **Volver al paso 3**

Arquitectura Q5: Ejercicio

(2)

Hacer un programa que arme un arreglo a partir de la celda 0099 solo con las posiciones pares de otro arreglo que comienza en 1099 y que finaliza con el valor 0



1 Arreglos

2 Modos indirectos