

# Organización de las Computadoras

2022

# Temario

1- Nuevo tema

2- Ejercicio tipo

# 1-1 Instrucciones de bifurcación

Todos los procesadores incluyen operaciones condicionales que permitan ejecutar una serie de instrucciones u otras en función de un resultado lógico.

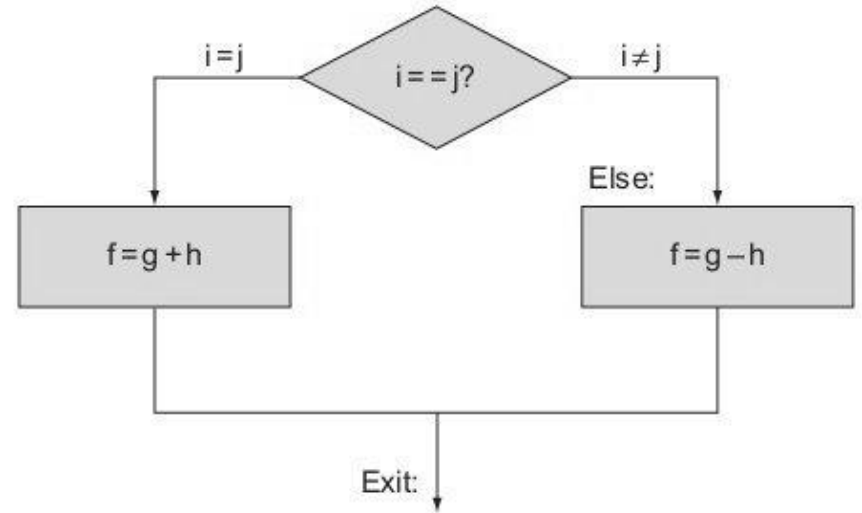
Esta toma de decisiones se representa en los lenguajes de programación mediante la sentencia if.

# 1-2 Ejemplo de un IF

Dado el siguiente código en lenguaje de alto nivel, codifique en assembler.

```
If (i==j) then  
    f = g+h;  
else  
    f = g-h;  
end if;
```

Suponga que f, g, h, i, j están cargados en los registros: x19, x20, x21, x22, x23.



## 1-2 Ejemplo de un IF

```
If (i==j) then  
    f = g+h;  
else  
    f = g-h;  
end if;
```

```
bne x22,x23,else  
add x19,x20,x21  
beq x0,x0,fin  
else: sub x19,x20,x21  
fin: ...
```

# 1-2 Ejemplo de un IF

```
If (i==j) then  
    f = g+h;  
else  
    f = g-h;  
end if;
```

*Notar que la condición es la opuesta  
que en alto nivel*

```
bne x22, x23, else  
add x19, x20, x21  
beq x0, x0, fin  
else: sub x19, x20, x21  
fin: ...
```

## 2- Solución ejercicio 1 guía 3

1. Copiar el código, hacer un seguimiento y indicar la funcionalidad

*.text*

*ori t0,zero,0x476  
ori t1,zero,0x552  
ori t2,zero,0x622*

*slt t6,t1,t0  
beq t6,zero,no1  
slt t6,t2,t0  
beq t6,zero,no2  
or a0,zero,t0  
j fin*

*no2: or a0,zero,t2  
j fin  
no1: slt t6,t2,t1  
beq t6,zero,no2  
or a0,zero,t1  
fin: ori a7,x0,10  
ecall*

a) Escribir línea a línea que acción está realizando

## 2- Solución ejercicio 1 guía 3

1. Copiar el código, hacer un seguimiento y indicar la funcionalidad

*.text*

*ori t0,zero,0x476*

*ori t1,zero,0x552*

*ori t2,zero,0x622*

*slt t6,t1,t0*

*beq t6,zero,no1*

*slt t6,t2,t0*

*beq t6,zero,no2*

*or a0,zero,t0*

*j fin*

*no2: or a0,zero,t2*

*j fin*

*no1: slt t6,t2,t1*

*beq t6,zero,no2*

*or a0,zero,t1*

*fin: ori a7,x0,10*

*ecall*

a) Escribir línea a línea que acción está realizando

Carga tres valores en t0, t1, t2.

Compara si t1 es menor que t0 y guarda el resultado en t6.

Bifurca si t6 es igual a cero (es decir bifurca si no es menor) a nro1.

Sino, Compara si t2 es menor que t0 y guarda el resultado en t6.

Bifurca si t6 es igual a cero (es decir bifurca si no es menor) a nro2.



## 2- Solución ejercicio 1 guía 3

1. Copiar el código, hacer un seguimiento y indicar la funcionalidad

*.text*

*ori t0,zero,0x476*

*ori t1,zero,0x552*

*ori t2,zero,0x622*

*slt t6,t1,t0*

*beq t6,zero,no1*

*slt t6,t2,t0*

*beq t6,zero,no2*

*or a0,zero,t0*

*j fin*

*no2: or a0,zero,t2*

*j fin*

*no1: slt t6,t2,t1*

*beq t6,zero,no2*

*or a0,zero,t1*

*fin: ori a7,x0,10*

*ecall*

Sino, Guarda t0 en a0. Y Salta a fin.

nro2: Guarda t2 en a0. Y Salta a fin.

nro1: Compara si t2 es menor que t0 y guarda el resultado en t6.

Bifurca si t6 es igual a cero (es decir bifurca si no es menor) a nro2.

Sino Guarda t1 en a0.

fin: Guarda 10 en a7, Llama al sistema

## 2- Solución ejercicio 1 guía 3

1. Copiar el código, hacer un seguimiento y indicar la funcionalidad

*.text*

*ori t0,zero,0x476*  
*ori t1,zero,0x552*  
*ori t2,zero,0x622*

*slt t6,t1,t0*  
*beq t6,zero,no1*  
*slt t6,t2,t0*  
*beq t6,zero,no2*  
*or a0,zero,t0*  
*j fin*

*no2: or a0,zero,t2*  
*j fin*  
*no1: slt t6,t2,t1*  
*beq t6,zero,no2*  
*or a0,zero,t1*  
*fin: ori a7,x0,10*  
*ecall*

b) Probar su funcionamiento cambiando los valores hexadecimales que están en las 3 primeras líneas

## 2- Solución ejercicio 1 guía 3

1. Copiar el código, hacer un seguimiento y indicar la funcionalidad

*.text*

*ori t0,zero,0x476*

*ori t1,zero,0x552*

*ori t2,zero,0x622*

*slt t6,t1,t0*

*beq t6,zero,no1*

*slt t6,t2,t0*

*beq t6,zero,no2*

*or a0,zero,t0*

*j fin*

*no2: or a0,zero,t2*

*j fin*

*no1: slt t6,t2,t1*

*beq t6,zero,no2*

*or a0,zero,t1*

*fin: ori a7,x0,10*

*ecall*

c) Trate de reemplazar las instrucciones de las líneas 6 y 7 por una sola que realice lo mismo. Repita esto para las líneas 8 y 9, y las líneas 14 y 15.

## 2- Solución ejercicio 1 guía 3

1. Copiar el código, hacer un seguimiento y indicar la funcionalidad

*.text*

*ori t0,zero,0x476*

*ori t1,zero,0x552*

*ori t2,zero,0x622*

*bge t1,t0,no1      # slt t6,t1,t0*  
*# beq t6,zero,no1*

*bge t2,t0,no2      # slt t6,t2,t0*  
*# beq t6,zero,no2*

*or a0,zero,t0*

*j fin*

*no2: or a0,zero,t2*  
*j fin*

*no1: bge t2,t1,no2      # slt t6,t2,t1*  
*# beq t6,zero,no2*

*or a0,zero,t1*  
*fin: ori a7,x0,10*  
*ecall*

c) Trate de reemplazar las instrucciones de las líneas 6 y 7 por una sola que realice lo mismo. Repita esto para las líneas 8 y 9, y las líneas 14 y 15.

**Preguntas ?**