Operaciones Aritméticas y Lógicas, e instrucciones de saltos condicionales

## **PREGUNTAS AUTOEVALUACIÓN:**

### **Módulo op\_arit\_log.s:**

* **Indicar cómo asociar el valor de los sumandos a las macros OPE1 y OPE2:**
* **Sin cambiar el valor de los operandos:**
  + **indicar el valor de la resta 1:** -15 (es almacenado en EAX)
  + **indicar el valor de la multiplicación 2:** -15 (es almacenado en AL)
  + **indicar el valor de la división 3:** 1 (es almacenado en AL)
  + **indicar el valor de la división 4:** 15 (es almacenado en AL)
  + **indicar el valor de las operaciones lógicas 5:** la primera (shr $4,%eax) desplaza el número en eax cuatro posiciones hacia la derecha (lo divide entre 2 elevado a la cuarta), la segunda (sar $4,%eax) hace lo mismo pero con extensión de signo (1 si es negativo, 0 si es positivo)

### **Módulo saltos.s:**

#### **Registro de flags:**

* **Indicar el contenido del registro EAX y el estado de los flags CF,ZF,SF,PF,OF después de la ejecución de cada instrucción :**
  + Antes del bloque: $eflags = [ PF ZF IF ] y EAX = 0
  + Después de *mov $0xFFFFFFFF,%eax*: $eflags = [ PF ZF IF ] y EAX=-1
  + Después de *shr $1,%eax*: $eflags = [ CF PF IF OF ] y EAX=2147483647
  + Después de *add %eax,%eax*: $eflags = [ AF SF IF OF ] y EAX=-2
  + Después de *testb $0xFF,%eax*: $eflags = [ SF IF ] y EAX=-2
  + Despues de *cmpl $0xFFFFFFFF,%eax*: $eflags = [ CF PF AF SF IF ] y EAX=-2

#### **Saltos:**

* **Indicar el estado de los flags CF,ZF,SF,PF,OF antes de la ejecución de la instrucción de salto e indicar si se produce o no el salto:**
  + Antes de los saltos $eflags = [ CF PF IF ]
  + Se produce el salto 2 (porque ZF está desactivada y ademas OF = SF) pero no el salto 1 (ya que CF está activada)

## **MEMORIA:**

set $eax = valor Para cambiar el valor de EAX

PF cuenta el número de unos que tiene el resultado, si es par es verdadero, sino es falso (banderín de paridad)

Los saltos condicionales se producen o no dependiendo del valor de los banderines

### Comandos para seguir el valor de los banderines:

layout regs

watch $eflags Se quita igual que los breakpoints (delete breakpoints)

### Que es cada banderín:

CF: El resultado de la operación tiene llevada del bit MSB del destino

OF: El resultado de la operación con signo se desborda, su tamaño supera el permitido.

ZF: el resultado de la operación tiene valor cero

SF: el resultado de la operación tiene valor negativo

PF: el resultado de la operación tiene el byte LSB con un número par de bits

### Tamaño de los registros en AMD64:

Registro RAX: es un registro de 64 bits

Registro EAX: son los 32 bits de menor peso de RAX

Registro AX: son los 16 bits de menor peso de RAX

Registro AL: son los 8 bits de menor peso de RAX

Registro AH: es el byte con los bits de las posiciones 8:15 de RAX

ALU solo puede operar con números enteros, para hacerlo con números reales se necesita la FPU (float process unit)