

Organización del Computador 2

Práctica 1a - Lenguaje Ensamblador IA-32, intel 64

- 1) Defina en lenguaje ensamblador los siguientes datos, con valores arbitrarios:
 - a) N1 y N2 Enteros sin signo de 8 bits
 - b) N3 y N4 Enteros de 16 bits
 - c) N5 y N6 Enteros de 32 bits
 - d) N7 y N8 Enteros sin signo de 64 bits
 - e) F1 Punto flotante de precisión simple
 - f) T Cadena de caracteres asciiz
- 2) Usando las definiciones del punto 1) escriba un programa que obtenga:
 - a) $N1 + N2$
 - b) $N3 - N4$
 - c) $N1 * N2$
 - d) Cociente y resto de la división entera $N3/N4$
 - e) $N5 + N6$
 - f) $N7 + N8$ usando registros de 32 bits
 - g) Cociente y resto de la división $N5/N6$ usando registros de 32 bitsReserve espacio en memoria de tamaño adecuado para almacenar cada resultado.
- 3) ¿Cómo podría resolver la suma de números de 128 bits, usando registros de 32 bits?
- 4) Usando la instrucción CMP y saltos condicionales, escriba un programa que muestre mensajes en pantalla indicando:
 - a) Si N1 es mayor que N2
 - b) Si el resultado de $N3 - N4$ es cero
 - c) Si $N5 + N6$ produce desbordamiento
 - d) Si N5 es negativo o positivo
 - e) Si $N7 + N8$ produce acarreo
 - f) Si el resultado de $N5 + N6$ tiene paridad par
- 5) Escriba un programa que muestre un mensaje indicando si el número F1 del punto 1) es cero o no, usando instrucciones lógicas (tener en cuenta que la representación en punto flotante del 0 tiene signo).
- 6) Escriba un programa que calcule la longitud de la cadena de caracteres T.
- 7) Escriba un programa que recorra una secuencia de números de 8 bits y encuentre el mayor.
- 8) Usando instrucciones lógicas, escriba un programa que determine si un número de 16 bits es múltiplo de 8.
- 9) Usando la instrucción pushfd, que apila el registro EFLAGS e instrucciones lógicas, escriba un programa que muestre en pantalla el estado de cada flag.