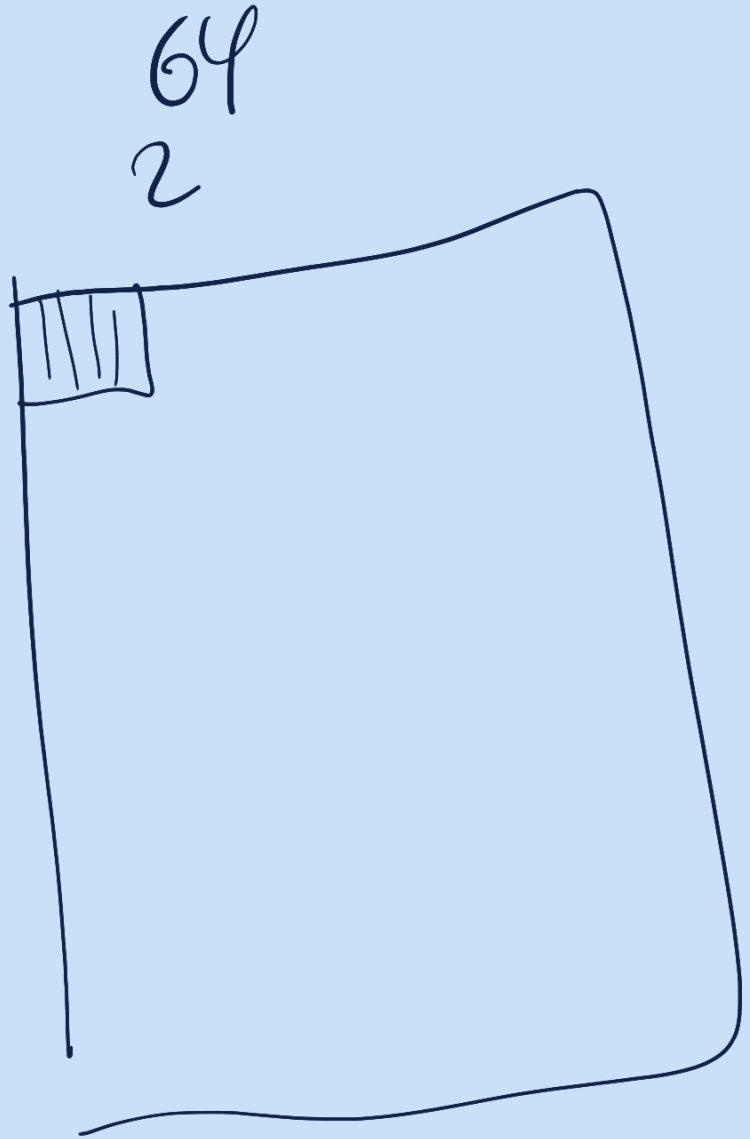


ES 1.

a)



TAMANO DE DIRECCION:
64 BITS



UNIDAD MINIMA:

1 BYTE

(C2U00)



b) 16 . 64 BITS CADA UNO.

c) TAMAÑO 64 BITS.

ALMACENA EL OFFSET HASTA
LA SIGUIENTE INSTRUCCION

TIENE ESE TAMAÑO PARA PODER
RECORRER TODA LA MEMORIA?

$\square \square \square = 0$

111

X		
		Y

2)

a) TAMAÑO 32 BIT
ALMACENA EL ESTADO DE
LAS FLAGS.

6) FLAG LENO: 6
FLAG CARRY: 0
FLAG INTERRUPT: 9

c) ?

3) a)

- PROCESOS Y SUBROUTINAS LLAMADOS (SABER A DONDE VOLVER)
- PASADA DE PARAMETROS ENTRE P.Y S.

0 UBICADA EN LA ULTIMA POSICION DE MEMORIA

a) ESP: INICIO DEL STACK POINTER

EBP: LUGAR AL QUE VOLVEMOS

CONSIDERACIONES: EL ANCHO / TAMAÑO
DE COMO GENERAMOS EL
SEGMENTO ACTUAL

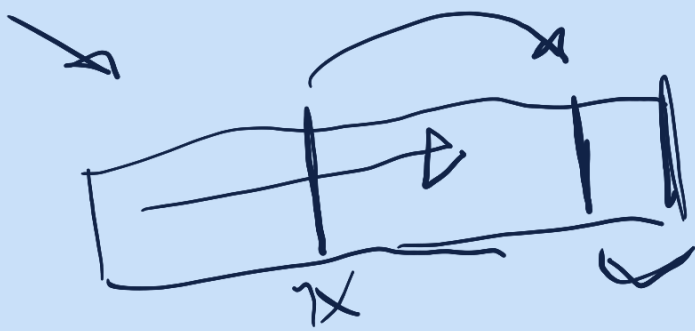


b) EL EIP, PARA SABER A
DONDE VOLVER AL TERMINAR
LA SUBROUTINA.

c) SE HACE POR DE LA
PIZA Y SE PONE SU
CONTENIDO EN EIP.

d) DEBE ASEGURARSE QUE EL
STACK SEA

FIRST OUT 32C
 la dirección correcta a
 retornar.



• SE PUEDEN ASSEGURAR
 QUE EL
 ES CORRECTO COPIANDO EL
 CONTENT DE EBP

¿NO PISAS EL EBP QUÉ?

e) 16 y 32 BITS

T. D. I

PARA MODOS 32 y 64

BITS RESPECTIVAMENTE

F) PODER, PODES HACER LO QUE QUIERAS.

OTRO: EL TAMAÑO DEL

RBP ES 32 BITS (64 MODE)

4)

INC

a) 1 OPERANDO

TIPOS:

r/m (8, 16, 32, 64)

r (16, 32)

b) INCREMENTA EL
REGISTRO EN 1.

c) INC EAX

SUB

a) 2 OPERANDOS

— REG, MEM

— IMM, REG, MEM

b) RESTA EL SEGUNDO
OPERANDO DEL PRIMERO
Y ALMACENA EN EL
SEGUNDO

c) SUB RAX, 0x0001

XOR

a) 2 OPERANDOS

— REG, MEM

— IMM, REG, MEM

b) HACE XOR ENTRE ESTOS
DOS Y LO GUARDA

EN EL DESTINO (PRIMER OPERANDO)

c) XOR EBX, ECX

JE

a) 1 OPERANDO

— RELATIVE OFFSET

b) BRANCHES SI ESTA
ENCENDIDA LA FLAG ZF

c) JE START

JZ

a) 1 OPERANDO

— RELATIVE OFFSET

b) BRANCH SI ZF ESTA

ENCLOSURE

d) 52 NO DISCONIBLE EN
64 BIT

