

P1 Si $SP = 0006h$ y $FLAGS = 0210h$ al inicio de la ejecución del código que se adjunta, indican los valores de las 6 primeras posiciones de la pila tanto cuando todos los procedimientos son NEAR como FAR; la pila está inicializada a cero

2100:2250 E8A8FD call teen - Datan
 2100:2253 89161000 mov Datan[0], dx

En el caso near se guarda únicamente el offset de retorno mientras que en FAR es offset y segmento

Near:

0 1 2 3 4 5	FAR	0 1 2 3 4 5
0 0 0 0 53 22		0 0 53 22 00 21

P2 Al inicio de la ejecución de una función invocada desde lenguaje C, se sabe que $SP = 2$ y que las 16 primeras posiciones de la pila son:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11h	A0h	25h	00h	32h	00h	A2h	E9h	00h	01h	24h	F1h	00h	03h	41h	12h

La sigla de dicha función es

int fun(char* p, long n);

indican el valor de los tres parámetros con que la función fue invocada tanto en near como far

Near $p = 0032h$

$n = C100E9A2h$

FAR $p = C100h$ E9A2h

$n = 16300 F124h$

P3 Escribir el código para poner 1 los bits 5, 10 y 14 del registro AX, dejando el resto inalterado y poner a 0 los bits 5, 10 y 14 de BX or AX, 01000100001000000b indican bits
 AND BX, 1011101111011111b

P5 Al inicio de la ejecución de una función invocada desde lenguaje C, se tiene que $SP=0$ y las 16 primeras posiciones de la tabla tienen estos valores

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
11h 40h 25h 00h 32h 00h 12h E9h 00h 01h 24h F1h 00h 63h 41h 12h

Indicar el valor de los parámetros con que la función fue invocada desde C int fun(char C, int n, char *p),

Near C = 0025h

n = 0032h

p = E9A2h

Far C = 0032h

n = E9A2h

p = F124h: C100h

P6 La función de lenguaje C cuya signature se indica en el recuadro de la derecha es invocada desde el programa de código máquina.

En el momento de la llamada se suponen los valores $SP=14$, $n=1234h$, $C=ABh$, $p=4253h:5678h$

4253:0007 E8F6F

call - Fun

4253:000A B8004C

mov ax, 4C00h

Fun(int n, char C, char *p)

Indicar el valor de las 16 posiciones iniciales de la pila al ejecutarse fun tanto en NEAR como FAR

FAR

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0A 00 53 42 34 12 AB 00 78 56 53 42

NEAR

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0A 00 34 12 AB 00 78 56

P7 Escriba en ensamblador la función - Multiply8U que multiplica 2 enteros sin signo de 8 bits mediante sumar sucesivas. El primer operando en BH y el segundo en BL, el resultado en A

- Multiply8U PROC NEAR

MOV AX, 0

MOV CX, 0 //contador de bucle

bucle:

CMP CL, BL

JE Fin //si ya se han

ADD AX, BH

INC CL

JMP bucle

Fin:

ret

- Multiply8U ENDP

P9 La función que se indica es invocada desde el programa código máquina de la pregunta sabiendo que $SP=16$, $n=4321h$, $C=12h$, $P=4234h$. 8765 Indicar el valor de las 16 primeras posiciones en Near o Far

Near
 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
 0Dh FFh 65h 87h 21h 43h 12h 00h
 S342:FF0A call-fun
 S342:FF0D mov ax, 4000h
 Far
 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
 00 FF 42 53 65 87 34 12 21 43 12h 00h
 Func (chr, p, int, char)

P10 Indicar el vector de la interrupción de impresión por pantalla

0000:0000 54 02 CF 15 CE 01 CF 15 04 00 70 00 D7 01 CF 15
 0000:0010 04 00 70 00 30 00 00 C8 30 00 00 C8 30 00 00 C8
 Imprimir pantalla INT 5 $5 \times 4 = 20 = 14h$
 Segmento: C800h OFFSET 00 30h

P11 Código para poner a 1 las bits 0, 7 y 14 de AX y a 0 las bits 2, 10 y 15 de BX
 or ax, 01000000010000001b
 and bx, 0111101111111011b

P13 Si $SP=0004h$ y $FLAGS=0200h$, indique las 6 primeras posiciones de la pila tras llamar

4250:025E E8A7FD call 8000-Data
 4250:0261 89161000 mov Data[0], dx

Near 0 1 2 3 4 5 6
 6102
 Far 0 1 2 3 4 5
 6102 5042

P14 Si $SP=0006h$ y $FLAGS=0200h$, muestre el valor de las primeras ocho posiciones de la pila

4250:025E C061 int 61h
 4250:0260 89161000 mov Data[0], dx
 0 1 2 3 4 5 6 7
 60 02 50 42 00 02

P18 Indicar el vector de la interrupción de punto de ruptura dado el siguiente volcado

0000:0000 54 02 CF 15 CE 01 CF 15 04 00 70 00 D7 01 CF 15
 0000:0010 04 00 70 00 30 00 00 C8 30 00 00 C8 30 00 00 C8
 Interrupción break point - INT 3h $3 \times 4 = 12 = C$

SEGMENTO 15CFh OFFSET 01D7h

Sabiendo que $SS=424Dh$, $SP=14$, $AX=3412h$ y $BX=5678$

Indicar el valor de las 16 primeras bytes del SS tras
 push AX
 pop BX
 push SS
 push BX
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
 12 34 4D 42

P21 Declarar mediante directivas en ensamblador las variables
 char nombre C20;
 short edad = 17;
 char tabla2D C10 C2;
 short valores C5 = {1, 2, 3, 4, 5};
 char despedida C20 = "Hasta luego";
 - nombre DB 20 dup(P)
 - edad DW 17
 - tabla DB 10*2 dup(P)
 - valores DW 1, 2, 3, 4, 5
 - despedida DB "Hasta luego", 20-(\$-despedida) dup(P)

P24 Al inicio de la ejecución de una función invocada desde lenguaje C, se tiene que SP=2 y que las 16 primeras direcciones de la pila son:
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
 11h 40h 25h 00h 32h 00h 42h 09h 00h c1h 24h F1h 00h 63h 41h 12h
 La sigla de dicha función es fun (char* p, long n).
 Indicar el valor de los parámetros en caso Near o Far
 Near P = 0032h n = C100E9A2h
 Far P = C100h: E9A2h n = 6300F124h

P25 Si SP=0004h y Flags=0210 indicar las 8 primeras direcciones de la pila al ejecutar el procedimiento Leer_Palos tanto en Near como Far
 20FF:4000 E8A8PD call Leer_Palos
 20FF:4003 89161000 mov Palos C0, dx

Near 0 1 2 3 4 5 No para parámetros, solo guarda dir de retorno
 03 40

Far 0 1 2 3 4 5
 FF 20 03 40

P26 Declarar mediante directivas en ensamblador las siguientes variables de C, ten en cuenta que los strings terminan en 0 y que un short ocupa 2 bytes
 char caracter = 'a';
 short edad;
 short tabla C100;
 char valores C5 = {1, 2, 3, 4, 5};
 char despedida C12 = "Hasta luego";
 - caracter DB 'a'
 - edad DW 1 dup(P)
 - tabla DW 100 dup(P)
 - valores DB '1', '2', '3', '4', '5'
 - despedida DB "Hasta luego", 0, 20-[\$-despedida] dup(P)

P23 Escribir en ensamblador un procedimiento descartar32 que decremente en dos unidades el valor de la variable de 32 bits cuya dirección recibe mediante los registros AX y BX como se muestra en el código

datos segment

contador dd 0FFFFFFFh

datos ends

...
mov ax, OFFSET contador

mov bx, SEG contador

call descartar32

descartar32 PROC FAR

Push bx, cx, guardamos los registros a usar

mov cx, bx

mov bx, ax

sub WORD PTR es:[bx], 2 restamos a los 2 primeros bytes 2

sbb WORD PTR es:[bx+2], 0 restamos el acarreo a los otros 2

Pop cx, bx recuperamos los registros usados

RET

descartar32 ENDP

P24 al iniciar la ejecución de una función se tiene $SP=2$ y la pila siguiente sabiendo que la notación de la función es $\text{fun}(\text{char} * p, \text{long } n)$, indicar el valor de los parámetros tanto en NEAR como FAR

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11h	40h	25h	00h	32h	00h	A2h	E9h	00h	C1h	24h	F1h	00h	63h	41h	24h	24h

Near $p = 0032h$ FAR $p = EC100h$; $A2E9h$
 $n = C100E9A2h$ $n = 6300F124h$

P25 Si $SP=0004h$ y $Flags=0210$ indicar las 6 posiciones de la pila al ejecutar la primera instrucción del procedimiento leer datos tanto en NEAR como FAR

20FF:4000 call leer_datos

20FF:4003 mov datos[0], dx

No se para anaqueletar por lo que solo se quedan la dirección de retorno

Near 0 1 2 3 4 5
03 40

Far 0 1 2 3 4 5
03 40 FF 20

Escribir en ensamblador un procedimiento lexico
(contar 4-48) que incrementa en 4 el valor de la variable
de 48 bits que recibe en registros AX y BX

datos segment

contador db 6 dup(0)

datos ends

mov ax, OFFSET contador

mov bx, SEG contador

call contar_4-48

contador_4-48 PROC FAR

push bx, cx

mov cx, bx

mov bx, ax

add WORD PTR es:[bx], 4

adc WORD PTR es:[bx+2], 0

adc WORD PTR es:[bx+4], 0

pop es, bx

RET

contador_4-48 ENDP

P.29) Si SP=000Ah y FLAGS=1234h al inicio de la ejecución del código, Indican el valor de las 16 \pm parámetros de la pila en el momento de ejecutar la 1^a instrucción de la rutina de servicio de interrupción 1Ch

549A:025F int 1Ch

549A:0260 mov Palas[0], dx

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
60 02 9A 54 34 12

30) Al inicio de la ejecución de una función invocada desde C se tiene la siguiente pila

de los parámetros como en NEAR cual es el valor SP=?

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
11 0A 25 00 32 00 A2 E9 00 C1 24 F1 00 63 41 12

Near p = E9 A2

c = 00 h

n = F124

Far

p = F124: C100

c = 00

n = 1241

31) Escribir en ensamblador un procedimiento cercano denominado -Instalar 61h que modifique el vector con la dirección de otro procedimiento denominado -ASI-61h. El valor anterior de un vector de interrupción en una variable dentro del procedimiento de instalar

-Instalar -61h Proc Near

jmp inicio
rsi dw PP, valor del vector anterior

Inicio

push ax, ~~ax~~, bx

mov ax, 0

mov ~~ax~~, 0

~~mov ax, 0~~

mov bx, SEG - ASI-61h

mov ax, es [61h*4]

mov cs:[rsi], ax

mov ax, es [61h*4 + 2] ; Guardar el anterior vector

mov cs:[rsi+2], ax

cli ; quitar interrupciones no es necesario, se puede poner SEG - ASI-61h directamente

mov es[61h*4], bx ; guardar el segmento

mov ax, OFFSET - ASI-61h

mov es[61h*4+2], ax ; guardar el offset

pop bx, ~~ax~~, bx

~~mov ax, 0~~

~~ret~~

-Instalar -61h endp

33) Si SP=0004h y FLAG=1234h indican el valor de los 6 primeros parámetros de la pila al ejecutar leer datos tanto en NEAR como FAR

43FF:25E3 call leer_datos

43FF:25E6 mov Palen CO], dx

Near 0 1 2 3 4 5
E6 25

Far 0 1 2 3 4 5
E6 25 FF 43

34) La función de lenguaje C cuya signature se adjunta es invocada desde el programa sabiendo que FLAG=1234h, SP=12, n=ABCDh, p=1234h: 8765h, c=EFh indican el valor de los 16 ± parámetros de la tabla tanto en Near como Far

43FF:25E3 call_fun

43FF:25E6 mov ax, 4000h

fun(int n, int *p, char c)

Near 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
E6 25 CD AB 65 87 EF 00

Far 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
E6 25 FF 43 CD AB 65 87 34 12 EF 00

38) Al inicio de la ejecución de una función invocada desde lenguaje C, se tiene que $SP=8h$, las 16 primeras posiciones de la pila son:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	A0	25	00	32	00	A2	E9	00	C1	24	00	63	00	41	12

ent fun (char p, int n) tanto Near como Far

Near $p=24h$
 $n=0063h$

Far $p=63h$
 $n=1241h$

39) Al inicio de la ejecución de una función invocada considerando que $SP=0$ y la pila es:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11h	A0	25	00	32	00	A2	E9	00	C1	24	00	63	00	41	12

si la función es fun (char p, int n) realicen tanto para Near como Far el valor de los parámetros

Near $p=00320025h$ Far $p=E9A20032h$
 $n=E9A2h$ $n=C100h$

40) Escriba en ensamblador de 80x86 la función máxima sabiendo que está compilado en modelo small

41) Suponiendo $SP=8$ indicar el valor de los registros

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
FF	A0	25	00	32	00	A2	E9	00	C1	24	F1	00	63	41	12

pop AX
 pop BX
 push AX
 pop CX
 pop DX

AX = C100h BX = F124h CX = C100h
 DX = 6300h