Suppose that you are given the following MASM data segment declarations:

```
.data
idNumber    BYTE     ?
status      WORD     0
list        DWORD    42 DUP(?)
count       DWORD    ?
```

The address of *idNumber* is **0x4E00**.

What is the hexadecimal address of the 14th element of list? 4E37h
What is the hexadecimal address of count? 4EABh

Assume that **LO** and **HI** have already been assigned as constants with **LO < HI**, and **x** has been declared as **DWORD** in the data segment. Also, Irvine's library is included, and***Randomize*** has already been called. Which of the following code fragments will assign to **x** a "random" integer in the range **[LO .. HI]**? Check all that apply.

☑

```
mov    eax,HI
sub    eax,LO
inc    eax
call   RandomRange
add    eax,LO
mov    x, eax
```

☐

```
mov    eax,HI
call   RandomRange
mov    x,eax
mov    eax,LO
call   RandomRange
sub    x,eax
```

☑

```
mov    eax,HI
mov    ebx,LO
dec    ebx
sub    eax,ebx
call   RandomRange
add    eax,LO
mov    x, eax
```

☐

```
push    LO
push    HI
call    RandomRange
pop     x
```

This information is provided so that you can answer answer questions 3 through 7.

The following is a partial *main* with a call to a procedure to calculate the factorial of integer argument *x*, and store the result (*x*!) in memory.

- The initial address of the top of the stack is **0x0A50**.
- The value at *x* is 12 (decimal).
- The address of *result* (DWORD) is **0x1200**.
- The address of *nextStep* is **0x2080**.
- The initial value in *ebp* is **0x3B**.

```
main    PROC
        push    OFFSET result
        push    x
        call    factorial
nextStep:
        ; ...
        exit
main    ENDP
```

```
factorial   PROC
        push    ebp
        mov     ebp,esp
        mov     eax,1
        mov     ecx,      A       ;value of parameter x
again:
        mul     ecx
        loop    again

        mov     edi,      B       ;address of result

        mov       C       ,eax    ;save the factorial in result
        pop     ebp

        ret       D
factorial   ENDP

END     main
```

Please select the text that should be placed in the blank space labeled A.

- 0x2080
- [ebp+12]
- [edi]
- [ebp+8]

Please select the text that should be placed in the blank space labeled B.

- [ebp+8]
- [edi]
- [ebp+12]

- [ ] 0x2080

Please select the text that should be placed in the blank space labeled C.

- [ ] 0x2080
- [x] [edi]
- [ ] [ebp+12]
- [ ] [ebp+8]

Please select the number that should be placed in the blank space labeled D.

- [ ] 4
- [ ] 16
- [x] 8
- [ ] 12
- [ ] 2

What hexadecimal number will ESP contain when the "mov eax,1" instruction is executed?

A40h

Based on the code given below, please show the ***decimal*** contents of the array after execution has completed.

Array Contents:

0x2200: 8

0x2204: 16

0x2208: 25

0x220C: 35

The address of ***array*** is `0x2200`. Show the (decimal) contents of ***array*** after execution returns to label ***retAdd***:

```
MAXSIZE = 4
.data
array   DWORD MAXSIZE DUP(?)
.code
main    PROC
        push  MAXSIZE
        push  8
        push  OFFSET array
        call  whatzit
retAdd:
        ; ...
        exit
main    ENDP
```

```
whatzit PROC
        push  ebp
        mov   ebp,esp
        mov   edi,[ebp+8]
        mov   eax,0
        mov   ebx,[ebp+12]
        mov   ecx,[ebp+16]
        mov   edx,0
fill:
        add   eax,ebx
        mov   [edi+edx],eax
        inc   ebx
        dec   eax
        add   edx,4
        loop  fill

        pop   ebp
        ret   12
whatzit ENDP
```

You may find it helpful to produce a table similar to the following example, filling in the blanks for each iteration of the loop.

eax	ebx	ecx	edx	edi

Which of the following postfix expressions corresponds to the given infix expression?

`(13 + 14 - 3 + 2) / 2 ^ 3`

- 13 14 + 3 - 2 + 2 3 ^ /
- 13 14 + 3 2 - + 2 3 ^ /
- 13 14 + 3 2 + - 2 3 ^ /
- 13 14 + 3 - 2 + 2 3 / ^