For the following segment, what is SIZEOF myChecker (in decimal - ignore the .0000 from Canvas)

.data  
myChecker   BYTE   12h,  
                   34h,  
                   56h,  
                   78h,  
                   90h



The \_\_\_\_\_\_\_\_\_ operator returns a value that is equivalent to multiplying the number of elements in a single data declaration by the size, in bytes, of a single element of a data declaration.

|  |  |
| --- | --- |
|  | TYPE |
|  | OFFSET |

|  |  |
| --- | --- |
|  | LENGTHOF |
|  | SIZEOF |

|  |  |
| --- | --- |
|  | PTR |

The \_\_\_\_\_\_\_\_\_ operator returns the distance in bytes, of a label from the beginning of its enclosing segment, added to the segment register.

|  |  |
| --- | --- |
|  | SIZEOF |
|  | PTR |

|  |  |
| --- | --- |
|  | TYPE |
|  | LENGTHOF |

|  |  |
| --- | --- |
|  | OFFSET |

Storing a string byte using string primitives increments/decrements which register?

|  |  |
| --- | --- |
|  | EDX |
|  | ESI |

|  |  |
| --- | --- |
|  | ESP |
|  | EDI |

Match the string primitive to its purpose.

Load string byte



Store string byte



Clear direction flag



Set direction flag



Suppose that you are given the following partial data segment, which starts at address offset 0x1000 :  
.data  
idArray WORD 3546, 1534, 12, 3481, 154, 6423  
x DWORD LENGTHOF idArray  
y DWORD SIZEOF idArray  
z DWORD TYPE idArray

*x* contains what value, in decimal? (Ignore the .0000 from Canvas)



Suppose that you are given the following partial data segment, which starts at address 0x0700 :  
.data  
idArray DWORD 1800, 1719, 1638, 1557, 1476, 1395, 1314, 1233, 1152, 1071, 990  
u DWORD LENGTHOF idArray  
v DWORD SIZEOF idArray  
  
What value does EAX contain after the following code has executed? (Ignore the .0000 that Canvas sticks on the end)  
mov   esi, OFFSET idArray  
mov   eax, [esi+8\*TYPE idArray]



Suppose that you are given the following partial data segment, which starts at address offset 0x1000 :  
.data  
idArray WORD 3546, 1534, 12, 3481, 154, 6423  
x DWORD LENGTHOF idArray  
y DWORD SIZEOF idArray  
z DWORD TYPE idArray  
  
What is the hexadecimal OFFSET of the number "12" in *idArray*?



Given the following array declaration:  
.data  
matrix   DWORD   50 DUP(10 DUP(?))  
  
If **matrix[0][0]** is the 0th sequentially stored BYTE in memory, which sequentially stored BYTE is the first byte corresponding to matrix[10][4]? (in decimal - ignore the .0000 from Canvas)



Assume that your program has access to the following data segment (starting at address 0x310):  
.data  
id       DWORD  7  
matrix   WORD   50 DUP(10 DUP(?))  
  
What is the hexadecimal address of matrix[7][3] (the 4th element of the 8th row)?



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

56 / (42 \* 2.6 \* 2) + (256 / (128 - 64)) \* 3 ^ 12

|  |  |
| --- | --- |
|  | 56 42 2.6 \* 2 \* / 256 128 64 - / 3 12 \* ^ + |
|  | 56 42 2.6 \* 2 \* 256 /128 64 - / 3 12 ^ \* + |

|  |  |
| --- | --- |
|  | 56 42 2.6 \* 2 \* / 256 128 64 - 3 12 / ^ \* + |
|  | 56 42 2.6 \* 2 \* / 256 128 64 - / 3 12 ^ \* + |

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

4 5 + 5 ^ 3 4 / -

|  |  |
| --- | --- |
|  | (4 + 5) ^ (5 - 3) / 4 |
|  | 4 + 5 ^ 5 - 3 / 4 |

|  |  |
| --- | --- |
|  | (4 + 5) ^ 5 - 3 / 4 |
|  | (4 + 5) ^ 5 / 3 - 4 |

Which of the following FPU manipulations corresponds to the given infix notation?

Z = (A + B - C) / D \* E

|  |  |
| --- | --- |
|  | finit fld    A fld    B fadd fld    C fsub fld    D fdiv fld    E fmul fstp |
|  | finit fld    A fld    B fadd fld    C fsub fld    D fmul fld    E fdiv fstp   Z |

|  |  |
| --- | --- |
|  | finit fld    A fld    B fsub fld    C fadd fld    D fdiv fld    E fmul fstp   Z |
|  | finit fld    A fld    B fadd fld    C fsub fld    D fdiv fld    E fmul fstp   Z |

3rd one wrong

The \_\_\_\_\_\_\_\_\_ operator returns the size, in bytes, of a single element of a data declaration.

|  |  |
| --- | --- |
|  | SIZEOF |
|  | OFFSET |

|  |  |
| --- | --- |
|  | TYPE |
|  | LENGTHOF |

|  |  |
| --- | --- |
|  | PTR |

MASM will throw an error when assembling the following data segment:

.data  
myChecker   BYTE   12h  
            BYTE   34h  
            BYTE   56h  
            BYTE   78h  
            BYTE   90h

|  |  |
| --- | --- |
|  | True |
|  | False |

Which of the following is the correct addressing formula for matrix index https://oregonstate.instructure.com/equation_images/M_%257Br%252Cc%257DMr,c?

|  |  |
| --- | --- |
|  | BaseAddress\:+\:elementsPerColumn\cdot\left[\left(c\cdot elementSize\right)+r\right] |
|  | BaseAddress\:+\:elementSize\cdot\left[\left(c\cdot elementsPerColumn\right)+r\right] |

|  |  |
| --- | --- |
|  | BaseAddress\:+\:elementSize\:\cdot\:\left[\left(r\:\cdot\:elementsPerRow\right)+c\right] |
|  | BaseAddress\:+\:elementsPerRow\cdot\left[\left(r\cdot elementsSize\right)+c\right] |

If the string direction flag is not set, string operations will move backward through the string.

  True

False

Suppose that you are given the following partial data segment:  
.data  
myPtrCheck   BYTE   12h, 34h, 56h, 78h,  
                    90h, ABh, CDh, EFh  
.code  
...  
mov   eax, DWORD PTR [myPtrCheck+2]

EAX contains what value, in hexadecimal?

AB907856h

0hAB907856

xAB907856

AB907856

0xAB907856

Given the following array declaration, how many bytes of memory does array *matrix* require? (in decimal - ignore the .0000 from Canvas)  
.data  
matrix   WORD   13 DUP(15 DUP(?))

**You Answered**



Given the following array declaration:  
.data  
matrix   DWORD   50 DUP(10 DUP(?))  
  
If **matrix[0][0]** is the 0th sequentially stored BYTE in memory, which sequentially stored BYTE is the first byte corresponding to matrix[3][7]? (in decimal - ignore the .0000 from Canvas)

**You Answered**



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

(13 + 14 - 3 + 2) / 2 ^ 3

**You Answered**

 13 14 + 3 - 2 + 2 3 / ^

13 14 + 3 - 2 + 2 3 ^ /

13 14 + 3 2 + - 2 3 ^ /

13 14 + 3 2 - + 2 3 ^ /

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

3 5 4 2 3 6 / \* - ^ +

  (3 + 5) ^ (4 - 2 \* 3 / 6)

  3 + 5 ^ (4 - 2 \* 3 / 6)

  3 / 5 \* (4 - 2 ^ 3 + 6)

3 + 5 ^ 4 - 2 \* 3 / 6

Which of the following infix notations corresponds to the given FPU manipulations? A B / C D - \* E -

finit  
fld    A  
fld    B  
fdiv  
fld    C  
fld    D  
fsub  
fmul  
fld    E  
fsub  
fstp   Z

  Y = A \* B / (C - D) - E

Y = A / B \* (C - D) - E

 Y = A / B \* C - D - E

Y = A / B \* (C - D - E)