

Ce valoare va fi stocata in variabila **s** in momentul in care executia va ajunge la **et_exit**? 2 points

```
.data
    v: .long 15, 21, 30, 16, 18, 12
    n: .long 6
    s: .long 0
.text

.global main

main:
    mov $v, %esi
    mov n, %eax
    shr $1, %eax
    mov $0, %ecx

et_loop:
    cmp %eax, %ecx
    jge et_exit
    movl (%esi), %ebx
    addl %ebx, s
    addl $8, %esi
    inc %ecx
    jmp et_loop

et_exit:
    movl $1, %eax
    movl $0, %ebx
    int $0x80
```

- ☒ 63
- ☐ 23
- ☐ 129
- ☐ 66



Fie urmatorul program. Este acesta scris corect? Daca da, de ce, daca nu, de ce? 2 points

```
.data
    x1: .long 5
    x2: .long 12
    x3: .long 27
    n: .long 3
    s: .long 0
    formatPrintf: .asciz "%d\n"
.text

.global main

main:
    movl $x1, %edi
    movl $0, %ecx
et_loop:
    cmp n, %ecx
    je et_exit

    movl (%edi, %ecx, 4), %eax
    addl %eax, s

    incl %ecx
    jmp et_loop
et_exit:
    push s
    push $formatPrintf
    call printf
    pop %ebx
    pop %ebx

    movl $1, %eax
    movl $0, %ebx
    int $0x80
```



- ☐ nu este scris corect, deoarece foloseste x1 pe post de array, in timp ce x1 este doar o variabila
- ☐ nu este scris corect, deoarece n trebuia sa fie egal cu 1, intrucat array-ul x1 este un array doar cu un element
- ☒ este scris corect, deoarece x1 e doar o adresa, ca numele unui vector
- ☐ este scris corect, dar nu va functiona conform asteptarilor - ruleaza, dar instructiunile nu au o logica specifica
- ☐ nu este scris corect, din cauza ca intram in zone de memorie in care nu avem drepturi



Fie codul urmator. Stiind ca **y** este un **.long 0** declarat in sectiunea **.data**, care dintre urmatoarele poate fi valoarea initiala din **x**, stiind ca in urma apelului **printf** se va afisa la STDOUT valoarea 1572?

2 points

```
main:
    movl $0, %ecx
    movl $10, %edi
et_while:
    cmp x, %ecx
    je et_exit

    movl x, %eax
    movl $0, %edx
    div %edi

    movl %eax, x
    movl %edx, %esi

    movl y, %eax
    mul %edi

    add %esi, %eax
    movl %eax, y
    jmp et_while

et_exit:
    push y
    push $formatStr
    call printf
    popl %ebx
    popl %ebx
```

- ☐ 1572
- ☐ 15720
- ☒ 275100
- ☐ -1572



- ☐ 1024
- ☐ 216
- ☐ 83412
- ☐ 972250
- ☒ 27510



Ce valori vor fi depozitate in v cand executia va ajunge in dreptul etichetei **et_exit** ?

2 points

```
.data
    v: .space 20
    n: .long 5
.text

.global main

main:
    movl $v, %edi
    mov $11, %edx
    mov $0, %ecx

et_loop:
    cmp n, %ecx
    jg et_exit
    mov %edx, (%edi, %ecx, 4)
    inc %ecx
    inc %edx
    jmp et_loop

et_exit:
    mov $1, %eax
    movl $0, %ebx
    int $0x80
```

- ☐ de la 11 la 16
- ☐ de la 11 la 15
- ☒ de la 11 la 27
- ☐ de la 0 la 5
- ☐ Executia nu ajunge la et_exit, loop-ul este infinit.



Fie urmatorul program. Ce valoare vom obtine daca vom rula cu debuggerul urmatoarele comenzi? 2 points

b et_exit

run

i r eax

```
.data
n: .long 4
v: .long 0x01020304, 0x05060708, 0x090a0b0c, 0xd0e0f10
.text
.global main
main:
    mov $v, %esi
    mov $2, %ecx
    mov -8(%esi, %ecx, 4), %eax
et_exit:
    mov $1, %eax
    xor %ebx, %ebx
    int $0x80
```

- ☒ 0x01020304
- ☐ 0x090a0b0c
- ☐ 0x05060708
- ☐ Executia nu va ajunge la et_exit, o sa apara o eroare de accesare

Clear selection

Page 2 of 2

Back

Submit

Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#) - [Privacy Policy](#).

Google Forms

