.data

x: .space 4

y: .space 4

v: .space 8

nrnoduri: .long 2

formatscanf: .asciz "%d"

formatprintf: .asciz "afisare: %d\n"

.text

.global main

main:

xor %ecx, %ecx

movl nrnoduri, %eax

mov $v, %edi

et\_for:

cmp %ecx, %eax #verif cond pt for

jle et\_afis

push $x

push $formatscanf

call scanf #cin>>x

pop %ebx

pop %ebxSS

movl x, %edx

movl %edx, (%edi, %ecx,4) #v[i]=x

incl %ecx #i++

jmp et\_for

et\_afis:

#restabilim registri de care avem nevoie pt afisarea vectorului

xor %ecx, %ecx

et\_a:

cmp %ecx, %eax #verif cond pt for

jle et\_exit

movl (%edi, %ecx, 4), %edx #scoatem valoarea din v[i]

movl %edx, x

push x

push $formatprintf

call printf #afisare v[i]

pop %ebx

pop %ebx

incl %ecx #i++

jmp et\_a

et\_exit:

movl $1, %eax

movl $0, %ebx

int $0x80