

bits 32

extern \_printf

global \_afisareASM

segment data public data use32

format db "%d ",0

sir1 times 10 dd 0

segment code public code use32

\_afisareASM:

push EBP

mov EBP,ESP

;pushad

;sub esp, 4 \* 3

;add ESP, 4\*1

mov ECX,[ESP+8]

mov ESI,[ESP+12]

lodsd

et1:

lodsd

push ECX

;printf("%d",EAX)

push EAX

push dword format

call \_printf

add ESP,4\*2

pop ECX

loop et1

mov EAX,0

;popad

mov ESP,EBP

pop EBP

ret

#include <stdio.h>

#include <stdlib.h>

int v[50];

int N[50];

int P[50];

int afisareASM(int n,int v[]);

int main() {

FILE\* f=fopen("numere.txt","r");

int x;

int ln=0;

while(fscanf(f,"%d",&x)!=EOF){

v[++ln]=x;

}

int n=0,p=0;

for(int i=1; i<=ln; ++i){

if(v[i]<0){

N[++n]=v[i];

} else {

P[++p]=v[i];

}

}

printf("Numere negative: ");

afisareASM(n,N);

printf("\n");

printf("Numere pozitive: ");

afisareASM(p,P);

return 0;

}

Graphical user interface, application

Description automatically generated with medium confidence Text

Description automatically generated Text

Description automatically generated