

## Test

### Grupa 3.2

Timp de lucru 20 minute

Timp de transmis fisierul sursa 5 minute pe adresa

[renata.boar@gmail.com](mailto:renata.boar@gmail.com)

Sursa va fi salvata cu

nume\_prenume\_grupa3\_2\_ **nrproblema**

unde **nrproblema** reprezinta numarul problemei asociat numelui vostru

Nr probl	ID	Grupa	Nume	obs
1.	LH66184	1A.03.2	GICIC IGOR	
2.	LH612381	1A.03.2	PASCARI GABRIEL	
3.	LH612297	1A.03.2	SAFTA M. ROBERT-ANDREI	
4.	LH612298	1A.03.2	SANDU I. MĂDĂLINA	
5.	LH612301	1A.03.2	SERACOVANU S. D. OANA-MARIA	
6.	LH612303	1A.03.2	SMĂDOI G. GEORGETA-MIHAELA	
7.	LH612304	1A.03.2	SOMOGYI E. O. CĂTĂLIN-CLAUDIU	
8.	LH612305	1A.03.2	STANA G. D. SABINA-ELENA	
9.	LH612308	1A.03.2	STOICA I. GHEORGHE-DENIS	
10	LH612336	1A.03.2	STOICA N. ESTERA-TABITA	
11	LH612330	1A.03.2	SUCEAVA A. M. GABRIEL-WILLIAM	
12	LH612310	1A.03.2	TOMA N. ANDREIA-IRINA	
13	LH612999	1A.03.2	ȘARI Z. F. JESSICA-AMALIA	
14	LH612300	1A.03.2	ȘCHIOPU E. V. ȘTEFANIA - TEODORA	
15	LH612307	1A.03.2	ȘTEFĂNESCU R. E. ANDREEA	
16	LH612309	1A.03.2	ȚIMBOTA-BELIN M. MIHAI	
17	LH612311	1A.03.2	ȚUȚA I. DANIELA	Joi ora 12
18	LH612350	1A.03.2	CIUMEICA ELENA	Miercuri ora 14

## Problemele

1.

Se citesc  $n$  ( $n \leq 30$ ) valori întregi, maxim 10 cifre fiecare, folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $v[i]$ ).

Să se afișeze, folosind doar pointeri, dublul tuturor valorilor negative.

Sa se insereze, după fiecare element cu valoarea 2 un element cu valoarea opusă (-2) și apoi să se afișeze noul sir, tot prin intermediul pointerilor.

Se vor construi funcții corepunzatoare.

2.

Se citesc de la tastatură cele maxim  $k$  ( $k \leq 50$ ) valori reale, ale unui vector  $x$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $x[i]$ ).

Sa se determine și să se afișeze media aritmetică a elementelor strict pozitive, folosind doar pointeri.

Sa se formeze, prin intermediul pointerilor, un nou sir  $v$ , care să conțină doar elementele pentru care valorile din sirul  $x$  sunt strict mai mici decât media aritmetică obținută anterior. Afișați noul sir.

Se vor construi funcții corepunzatoare.

3.

Se citesc  $m$  ( $m \leq 20$ ) valori întregi, maxim 20 cifre fiecare, reținute în vectorul  $a$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $a[i]$ ).

Să se afișeze, folosind doar pointeri, elementul maxim din  $a$ .

Sa se determine cea mai lungă secvență formată din elemente egale existente în vectorul  $a$ . Afișați lungimea secvenței, poziția de început și valoarea care se repetă, tot prin intermediul pointerilor.

Se vor construi funcții corepunzatoare.

4.

Se citesc  $n$  ( $n \leq 100$ ) valori naturale, maxim 10 cifre fiecare, reținute în vectorul  $b$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $b[i]$ ).

Să se afișeze, folosind doar pointeri, numărul elementelor din  $b$ , pentru care prima cifră e mai mică decât ultima cifră la fiecare element.

Sa se insereze, după fiecare element frecvența sa de apariție în sir iar apoi să se afișeze noul sir, tot prin intermediul pointerilor.

Se vor construi funcții corepunzatoare.

5.

Realizați un program care citește de la tastatură un text pe mai multe linii, terminat cu linie vidă.

Să se extragă cuvintele din text, să se memoreze, să se sorteze alfabetic și să se afișeze.

Se vor construi funcții corepunzatoare.

**6.**

Se citesc  $n$  ( $n \leq 40$ ) valori întregi, maxim 10 cifre fiecare, reținute în vectorul  $v$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $v[i]$ ).

Să se afișeze, folosind doar pointeri, elementul minim din  $v$ .

Sa se determine cea mai lunga secventa formata din elemente egale existente în vectorul  $v$ . Afisati lungimea secventei, pozitia de sfarsit si valoarea care se repeta, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare.

**7.**

Se citesc de la tastatura cele maxim  $p$  ( $p \leq 700$ ) valori reale, ale unui vector  $v$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $v[i]$ ).

Sa se determine si sa se afiseze media aritmetica a elementelor strict negative, folosind doar pointeri.

Sa se formeze, prin intermediul pointerilor, un nou sir  $a$ , care sa contina doar elementele pentru care valorile din sirul  $v$  sunt strict mai mari decat media aritmetica obtinuta anterior. Afisati noul sir.

Se vor construi functii corepunzatoare.

**8.**

Se citesc  $n$  ( $n \leq 200$ ) valori naturale, maxim 20 cifre fiecare, reținute în vectorul  $z$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $z[i]$ ).

Să se afișeze, folosind doar pointeri, numărul elementelor din  $z$ , pentru care prima cifra e mai mare decat ultima cifra la fiecare element.

Sa se insereze, dupa fiecare element, frecventa sa de aparitie în sir iar apoi sa se afiseze noul sir, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare.

**9.**

Realizati un program care citește de la tastatura un text pe mai multe linii, terminat cu linie vida.

Să se extraga cuvintele din text, sa se memoreze, sa se sorteze alfabetic si sa se afiseze.

Se vor construi functii corepunzatoare.

**10.**

Se citesc de la tastatura cele maxim  $n$  ( $n \leq 50$ ) valori reale, ale unui vector  $v$ , folosind doar pointeri (fără nici un fel de indexare - înseamnă că în cod nu va exista nici un  $v[i]$ ).

Sa se determine si sa se afiseze media aritmetica a elementelor strict pozitive, folosind doar pointeri.

Sa se formeze, prin intermediul pointerilor, un nou sir  $t$ , care sa contina doar elementele pentru care valorile din sirul  $v$  sunt strict mai mici decat media aritmetica obtinuta anterior. Afisati noul sir.

Se vor construi functii corepunzatoare.

**11.**

Se citesc  $m$  ( $m \leq 20$ ) valori întregi, maxim 20 cifre fiecare, reținute în vectorul  $a$ , folosind doar pointeri(fără nici un fel de indexare- înseamnă că în cod nu va exista nici un  $a[i]$ ).

Să se afișeze, folosind doar pointeri, elementul maxim din  $a$ .

Sa se determine cea mai lunga secventa formata din elemente egale existente in vectorul  $a$ . Afisati lungimea secventei, pozitia de inceput si valoarea care se repeta, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare.

**12.**

Se citesc  $n$  ( $n \leq 100$ ) valori naturale,maxim 10 cifre fiecare, reținute în vectorul  $p$ , folosind doar pointeri(fără nici un fel de indexare- înseamnă că în cod nu va exista nici un  $p[i]$ ).

Să se afișeze, folosind doar pointeri, numarul elementelor din  $p$ , pentru care prima cifra e mai mica decat ultima cifra la fiecare element.

Sa se insereze, dupa fiecare element frecventa sa de aparitie in sir iar apoi sa se afiseze noul sir, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare.

**13.**

Se citesc  $p$  ( $p \leq 245$ ) valori întregi,maxim 10 cifre fiecare, folosind doar pointeri(fără nici un fel de indexare- înseamnă că în cod nu va exista nici un  $x[i]$ ).

Să se afișeze, folosind doar pointeri, dublul tuturor valorilor pozitive.

Sa se insereze, dupa fiecare element cu valoare para un element cu valoarea opusa si apoi sa se afiseze noul sir, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare.

**14.**

Se citesc  $n$  ( $n \leq 40$ ) valori întregi,maxim 20 cifre fiecare, reținute în vectorul  $v$ , folosind doar pointeri(fără nici un fel de indexare- înseamnă că în cod nu va exista nici un  $v[i]$ ).

Să se afișeze, folosind doar pointeri, elementul minim din  $v$ .

Sa se determine cea mai lunga secventa formata din elemente egale existente in vectorul  $v$ . Afisati lungimea secventei, pozitia de sfarsit si valoarea care se repeta, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare.

**15.**

Realizati un program care citeste de la tastatura un text pe mai multe linii, terminat cu linie vida.

Să se extraga cuvintele din text, sa se memoreze, sa se sorteze alfabetic si sa se afiseze.

Se vor construi functii corepunzatoare.

**16.**

Se citesc  $n$  ( $n \leq 200$ ) valori naturale,maxim 10 cifre fiecare, reținute în vectorul  $r$ , folosind doar pointeri(fără nici un fel de indexare- înseamnă că în cod nu va exista nici un  $r[i]$ ).

Să se afișeze, folosind doar pointeri, numarul elementelor din  $r$ , pentru care prima cifra e mai mare decat ultima cifra la fiecare element.

Sa se insereze, dupa fiecare element frecventa sa de aparitie in sir iar apoi sa se afiseze noul sir, tot prin intermediul pointerilor.

Se vor construi functii corepunzatoare