

Karataev Yury

Grupa 162

Programarea calculatoarelor

Nr. 2
28.01.2025

Temp de lucru: 2 ore.

Se acordă un punct din oficiu.

1.a) (0.5p) Reprezentați pe 16 biți, în baza 2, numerele întregi 294 și -167.

1.b) (0.5p) Determinați pas cu pas ce valori vor avea variabilele x și n în urma executării următoarei secvențe de cod:

```
int x=(10/10)*3;  
int n=100; n=n<<2;
```

1.c) (0.5p) Scrieți care este efectul executării următoarei secvențe de cod:

```
for (int i=1; i<=5; i++) if ((i%2)) puts("I");  
for (int i=1; i<=5; i++) for (int j=1; j<=5; j++) printf("%d ", i+j);  
int v[]={12,3,5,-6,7,10}; *p,*q;  
p=&v[0]; q=&v[3];  
printf("a%d b%d c%d d%d", v[1], *v, *v+4, *(v+4), *(v+4)-*(v+2));
```

1.d) (0.5p) Scrieți ce se afișează la rularea programului următor:

```
#include <stdio.h>  
#define R(x) x+2  
#define T(x) (x+2*x)  
int main ()  
{ int a=3; printf("a%d %d", R(a+1), a);  
  a=3; printf("a%d %d", 2*R(a+1), a);  
  a=3; printf("a%d %d", T(a-1), a);  
  a=3; printf("a%d %d", T(a++), a);  
  return 0;}
```

2. (0.75p) Scrieți un program care citește de la tastatură numele unui fișier text în care sunt scrise mai multe caractere, apoi citește caracter cu caracter din fișier și scrie pe ecran, pe câte o linie, separate prin spațiu, în ordinea în care au fost citite, fiecare caracter cifră și codul său ASCII. Dacă nu se scrie nici un caracter, se va scrie mesajul "Nu exista".

3. (1p) Scrieți o funcție care primește ca parametru un număr natural n ($n > 2$) și returnează un tablou bidimensional triunghiular superior alocat dinamic și construit astfel:
- prima coloană conține numerele de la 1 la n , în ordine crescătoare;
 - orice alt element este egal cu produsul dintre numărul precedent pe linie și numărul coloanei pe care se află.

Exemplu: pentru $n=4$ funcția va returna tabloul:

```
1 1 2 6
2 2 4
3 3
4
```

4. (0.75p) Scrieți definiția și apoi un exemplu de apel al unei funcții cu număr variabil de parametri care calculează suma unui șir de n numere întregi.

5. (0.75p) Două tablouri unidimensionale sunt egale dacă au același număr de elemente și elementele aflate pe aceeași poziție sunt egale. Scrieți definiția unei funcții generice f care să verifice dacă două tablouri unidimensionale sunt egale sau nu, apoi dați două exemple de apel pentru funcția f .

6. Scrieți secvențe de cod C în care:

- a) (0.25p) definiți o structură **student** care să permită memorarea numelui unui student (cel mult 35 de caractere, cuvinte și spații), a celor 5 note (întregi) obținute la examene, a numărului de credite (fiecare examen promovat cu cel puțin nota 5 are 5 credite) pe care acesta le-a obținut în sesiune, precum și a situației la finalul sesiunii (integralist/restanțier).
- b) (0.5p) scrieți o funcție care să calculeze numărul de credite și, în funcție de acesta, situația fiecărui dintre cei n studenți ale căror date sunt memorate într-un tablou unidimensional t cu elemente de tip **student**.
- c) (1p) folosind funcția **qsort** din biblioteca **stdlib.h**, sortați elementele unui tablou unidimensional t format din n elemente de tip **student** în ordinea descrescătoare a numărului de credite obținute. (Definiți funcția comparator corespunzătoare și scrieți apelul funcției **qsort**.)

7. (2p) Fișierul **text.txt** conține mai multe linii de text. Pe prima linie este scris w , un cuvânt cu cel mult 20 de caractere, iar orice altă linie din fișier conține cel mult 500 de caractere: cuvinte separate prin câte un spațiu.

Scrieți un program care verifică dacă w se află printre cuvintele unei linii și scrie în fișierul binar **linie.txt** numărul de ordine al fiecărei linii pe care se află w și numărul total de cuvinte de pe linia respectivă. Dacă nici o linie de text din fișierul cu date de intrare nu conține cuvântul w , atunci se va scrie în fișierul **linie.txt** valoarea 0.