

PC - testul 1

1. (5p) Scrieți în grila alăturată, în ordine, cele 10 valori care se afișează la ieșirea standard, pe rânduri separate, în urma execuției programului C de mai jos?

<pre>#include <stdio.h> int main() { int a = 0; double b = 12.0/a; printf("%f\n", b/b); printf("%s\n", (&a==0)?"TEST":"CURS"); char d = 29; char e = 100; unsigned char f = d+e; printf("%d\n", f); float g = d/e; printf("%f\n",g); double t[5]={1.11, 2.22, 3.33}; printf("%d\n", sizeof(t)); printf("%f\n", t[1]); int w=2; char s[]="321.1254,3f5"; d = sscanf(s,"%lf,%c%d",&b,&e,&w); printf("%c\n",d+e+w); int i, j, x=50; for (i=5;i<40;i++) for (j=30; j<60; j++) { if (i-j) continue; x+=3; } printf("%d\n",x); printf("%x\n", -1<<13); while (e--); ++a; printf("%d\n", e-a); return 0; }</pre>	Rezultate afișate (câte unul pe linie):

2. (4p) Scrieți un program C care citește mai întâi de la intrarea standard o secvență de numere naturale pozitive care nu conține duplicate. Numerele sunt separate printr-un singur spațiu, iar secvența se termină cu caracterul #. Se presupune că secvența nu poate avea mai mult de 1000 de numere și că datele de intrare sunt valide. Programul trebuie apoi să determine și să afișeze la ieșirea standard acele perechi de numere din secvență ale căror sumă este tot un număr care apare în secvență (ordinea de afișare a perechilor nu este importantă). În cazul în care nu există astfel de perechi se va afișa mesajul „Nu există!”.

Exemple:

Intrare:	Ieșire:
10 15 8 31 7 23 40 #	15+8=23 8+7=15 8+23=31
20 13 40 12 #	Nu exista!