

Care este numărul **minim** de muchii ce pot fi eliminate și care sunt aceste muchii astfel încât graful partial obținut să nu mai fie conex? **(6p.)**

## Lucrare de control – XI B, 20 ianuarie 2010, Rândul 2

4. Se consideră graful orientat cu 6 noduri reprezentat prin matricea de adiacență alăturată. Care este numărul tuturor grafurilor parțiale distincte ale grafului dat? Două grafuri parțiale sunt distincte dacă matricele lor de adiacență sunt diferite. (6p.)
- |   |   |   |   |   |   |
|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 |
5. Se consideră un text cu maximum 255 de caractere, format din litere mici ale alfabetului englez și spații. Textul conține cel puțin o consoană. Scrieți un program C/C++ care citește de la tastatură textul și apoi determină transformarea acestuia, eliminând numai ultima consoană care apare în text, ca în exemplu. Programul va afișa pe ecran textul obținut.
- Exemplu:** dacă de la tastatură se introduce textul: `mare frig saci`  
pe ecran se va afișa: `mare frig sai` (10p.)

### Subiectul III (30 de puncte)

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Generarea tuturor cuvintelor de trei litere mici, nu neapărat distincte, ale alfabetului englez, se poate realiza cu ajutorul unui algoritm echivalent cu cel de generare a: (4p.)
- produsului cartezian
  - combinărilor
  - aranjamentelor
  - permutărilor

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Subprogramul `f` este definit alăturat. (4p.)  
Ce valoare are `f(8, 4)`?
- ```
int f (int x,int y)
{
    if(x<y) return 1+f(x+1,y);
    if(y<x) return 1+f(y+1,x);
    return 1;
}
```

3. Scrieți un program C/C++ care citește de la tastatură un număr natural `n` din intervalul `[2, 50]` și apoi `n` numere reale și afișează pe ecran câte dintre cele `n` numere reale sunt egale cu media aritmetică a celorlalte `n-1` numere reale. (6p.)

4. a) Scrieți definiția completă a unui subprogram `primul`, care  
- primește prin singurul său parametru, `a`, o valoare naturală din intervalul `[2, 10000]`  
- returnează o valoare naturală reprezentând cel mai mic divizor al numărului `a` mai mare strict decât 1. (6p.)

b) Fișierul text `NUMERE.IN` conține pe prima linie un număr natural nenul `n` ( $1 \leq n \leq 100$ ) și pe următoarea linie `n` numere naturale din intervalul `[2, 10000]` separate prin câte un spațiu.

Un număr natural `n` se numește „aproape prim” dacă este egal cu produsul a două numere prime distincte. De exemplu, numărul 14 este „aproape prim” pentru că este egal cu produsul numerelor prime 2 și 7.

Scrieți un program C/C++ care determină, folosind apeluri utile ale subprogramului `primul`, cel mai mare număr „aproape prim” de pe linia a doua a fișierului `NUMERE.IN`. În cazul în care există un astfel de număr se afișează pe ecran mesajul `DA`, urmat de numărul determinat, iar în caz contrar mesajul `NU`.

**Exemplu:** dacă fișierul `NUMERE.IN` are conținutul:

6

100 14 21 8 77 35

atunci se afișează pe ecran `DA 77` pentru că numărul 77 este cel mai mare dintre numerele „aproape prime” din fișier ( $14=7*2$ ,  $21=7*3$ ,  $77=7*11$ ,  $35=7*5$ ). (10p.)