

2. Utilizând metoda backtracking, se generează toate buchetele formate din câte trei flori distincte din mulțimea ordonată astfel: {trandafir, crin, gerbera, iris, eustoma, orhidee, zambilă}; în același buchet crinul nu poate fi împreună cu eustoma sau cu zambila. Două buchete diferă prin cel puțin o floare. Primele cinci soluții sunt, în ordinea generării: (trandafir, crin, gerbera), (trandafir, crin, iris), (trandafir, crin, orhidee), (trandafir, gerbera, iris), (trandafir, gerbera, eustoma). Scrieți două soluții, una generată imediat înainte, iar cealaltă generată imediat după (crin, gerbera, orhidee). (6p.)
3. Gama muzicală conține, în această ordine, notele **do, re, mi, fa, sol, la, si**. Două note formează un interval muzical, iar acesta este numit **terță** dacă, în ordinea din gamă, între cele două note există o singură altă notă muzicală. În notația englezescă se folosesc unele litere mari ale alfabetului pentru notele muzicale, astfel: litera **A** pentru nota **la**, litera **B** pentru nota **si**, apoi, în ordine, litera **C** pentru nota **do**, litera **D** pentru nota **re**, litera **E** pentru nota **mi**, litera **F** pentru nota **fa** și litera **G** pentru nota **sol**. Variabila **m**, declarată alăturat, memorează literele corespunzătoare notației englezești pentru două note care formează un interval muzical, date în ordinea din gamă. Știind că în câmpul **nota1** se memorează una dintre literele **C, D, E, F** sau **G**, scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran mesajul **TERTA**, dacă intervalul respectiv formează o terță, sau mesajul **NU**, în caz contrar.
- ```
struct interval
{ char nota1;
 char nota2;
}m;
```
- Exemplu:** dacă în câmpul **nota1** se memorează litera **G** (pentru nota **sol**), iar în câmpul **nota2** se memorează litera **B** (pentru nota **si**), se afișează mesajul **TERTA**. (6p.)

### SUBIECTUL al III-lea

(30 de puncte)

1. Un număr natural nenul, **n**, se numește **moderat** dacă este egal cu produsul a două numere prime, iar acestea sunt consecutive în șirul numerelor prime (2, 3, 5, 7, 11, 13, 17...). Subprogramul **moderat** are un singur parametru, **n**, prin care primește un număr natural ( $n \in [1, 10^5]$ ). Subprogramul returnează valoarea 1, dacă **n** este un număr moderat, sau valoarea 0, în caz contrar. Scrieți definiția completă a subprogramului.
- Exemplu:** dacă **n=35**, subprogramul returnează 1 ( $35=5 \cdot 7$ ), iar dacă **n=18** sau **n=55** sau **n=4**, subprogramul returnează 0. (10p.)
2. O grădină este împărțită în parcele pătrate identice, dispuse pe rânduri succesive. Toate parcelele de pe un rând conțin același tip de plante (doar flori sau doar gazon) și oricare două rânduri alăturate conțin tipuri diferite de plante, **pe primul rând fiind gazon**. Pentru fiecare parcelă se memorează înălțimea acesteia, egală cu înălțimea medie a plantelor conținute, exprimată în decimetri. Pentru a pune în valoare florile, fiecare parcelă cu flori trebuie să fie mai înaltă cu cel puțin un decimetru decât oricare dintre parcelele cu gazon de pe rândul vecin anterior. S-a hotărât tunderea parcelelor cu gazon prea înalte, astfel încât ele să fie aduse la înălțimea maximă permisă. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul  $[2, 10^2]$ , **m** și **n**, apoi elementele unui tablou bidimensional cu **m** linii și **n** coloane, numere naturale din intervalul  $[2, 10]$ , reprezentând înălțimile parcelelor, în ordinea dispunerii acestora în grădină, rând după rând, și pe fiecare rând de la stânga la dreapta. Programul modifică apoi tabloul în memorie corespunzător situației terenului după tunderea gazonului și afișează pe ecran tabloul obținut, fiecare linie a tabloului pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.
- Exemplu:**  
pentru **m=4, n=6** și tabloul
- |   |   |   |   |   |   |
|---|---|---|---|---|---|
| 6 | 2 | 3 | 7 | 9 | 4 |
| 7 | 8 | 5 | 9 | 8 | 5 |
| 8 | 7 | 2 | 6 | 2 | 9 |
| 7 | 2 | 5 | 9 | 3 | 8 |
- se obține tabloul
- |   |   |   |   |   |   |
|---|---|---|---|---|---|
| 4 | 2 | 3 | 4 | 4 | 4 |
| 7 | 8 | 5 | 9 | 8 | 5 |
| 1 | 1 | 1 | 1 | 1 | 1 |
| 7 | 2 | 5 | 9 | 3 | 8 |
- (10p.)
3. Șirul 0, 0, 1, 4, 13, 38, 105, 280, 729 . . . este definit astfel:  $f_1=f_2=0, f_3=1, f_n=4 \cdot f_{n-1}-3 \cdot f_{n-2}-2 \cdot f_{n-3}$  (unde **n** este un număr natural  $n \geq 4$ ). Se citesc de la tastatură trei numere naturale **x, y** și **z** ( $x \leq y < z \leq 10^9$ ), valorile a trei termeni aflați pe poziții consecutive în șirul dat, și se cere să se scrie în fișierul **bac.txt**, în ordine descrescătoare, separați prin câte un spațiu, toți termenii șirului care sunt mai mici sau egali cu **z**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.
- Exemplu:** dacă se citesc numerele 38 105 280 fișierul conține numerele 280 105 38 13 4 1 0 0
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)