

**Examenul de bacalaureat național 2019**  
**Proba E. d)**  
**Informatică**  
**Limbajul Pascal**

**Varianta 2**

*Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică*  
*Filieră vocațională, profil militar, specializare matematică-informatică*

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- Identificatorii utilizați în rezolvări trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

**SUBIECTUL I**

**(20 de puncte)**

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

- O expresie Pascal care are valoarea false este:
  - 'm' < 'n'
  - 'm' = 'M'
  - 'M' = chr(ord('m') + ord('N') - ord('n'))
  - 'N' = chr(ord('M') + 1)
- Subprogramul `f` este definit alături. Indicați ce se afișează în urma apelului de mai jos.  
`f(75, 30);`

```
procedure f(i, j: integer);  
begin write(i, ' ');  
      if i <> j then  
        begin if i < j then  
              begin i := i + j; j := i - j; i := i - j end;  
              f(i - j, j)  
            end  
          end;  
end;
```

  - 75 30 45
  - 75 45 15
  - 75 45 15 15
  - 75 30 45 15
- Utilizând metoda backtracking, se generează toate torturile formate din trei straturi de tipuri distincte de cremă din mulțimea {caramel, ciocolată, frișcă, nuci, vanilie}. Ultimul strat este de frișcă sau de vanilie, iar aceste tipuri de cremă nu pot apărea pe primele două straturi. Două torturi cu straturi din aceleași tipuri de cremă sunt diferite dacă acestea sunt dispuse în altă ordine. Primele patru soluții generate sunt, în această ordine: (caramel, ciocolată, frișcă), (caramel, ciocolată, vanilie), (caramel, nuci, frișcă), (caramel, nuci, vanilie). A cincea soluție este:
  - (ciocolată, caramel, frișcă)
  - (ciocolată, nuci, frișcă)
  - (nuci, caramel, vanilie)
  - (nuci, ciocolată, vanilie)
- Numărul de noduri ale unui arbore cu 4 muchii este:
  - 2
  - 3
  - 4
  - 5
- Valorile care pot reprezenta gradele nodurilor unui graf neorientat, cu 6 noduri, sunt:
  - 1, 0, 0, 2, 2, 2
  - 2, 2, 3, 4, 0, 3
  - 2, 2, 5, 5, 0, 1
  - 6, 5, 4, 3, 2, 1

**SUBIECTUL al II-lea**

**(40 de puncte)**

**1. Algoritmul alăturat este reprezentat în pseudocod.**

S-a notat cu  $a \% b$  restul împărțirii numărului natural  $a$  la numărul natural nenul  $b$  și cu  $[c]$  partea întregă a numărului real  $c$ .

- Scrieți valoarea afișată dacă se citește numărul 27102. **(6p.)**
- Scrieți trei numere distincte din intervalul  $[10, 9999]$  care pot fi citite, astfel încât în urma executării algoritmului, pentru fiecare dintre acestea, valoarea afișată să fie 11. **(6p.)**
- Scrieți programul Pascal corespunzător algoritmului dat. **(10p.)**
- Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind prima structură `cât timp...execută` cu o structură de tip `pentru...execută`. **(6p.)**

```

citește n
  (număr natural)
m ← 0; x ← 1
cât timp x ≤ 9 execută
  cp ← n
  cât timp cp ≠ 0 execută
    dacă cp % 10 = x atunci
      m ← m * 10 + x
    cp ← [cp / 10]
  x ← x + 1
scrie m
  
```

- Fiind date două șiruri de caractere  $a$  și  $b$ , îl numim pe  $a$  **sufix** al lui  $b$  dacă  $a$  este egal cu  $b$  sau dacă  $b$  se poate obține din  $a$  prin alipirea la stânga a unor noi caractere. Variabilele  $a$  și  $b$  pot memora câte un șir cu cel mult 20 de caractere. Scrieți o secvență de instrucțiuni în urma executării căreia variabila  $a$  să memoreze un sufix al lui  $b$  format din trei caractere, sau șirul **nedeterminat**, dacă nu există un astfel de sufix. **Exemplu:** dacă  $b$  memorează șirul **centaur**, atunci  $a$  memorează șirul **aur**, iar dacă  $b$  memorează șirul **au**, atunci  $a$  memorează șirul **nedeterminat**. **(6p.)**

- În declarațiile alăturate, variabila  $p$  memorează coordonatele unui punct în sistemul de coordonate  $xOy$ , iar variabila  $c$  memorează datele caracteristice ale unui cerc: lungimea razei și coordonatele centrului său. Scrieți o expresie care are valoarea **true** dacă și numai dacă punctul corespunzător variabilei  $p$  coincide cu centrul cercului corespunzător variabilei  $c$ . **(6p.)**

```

type punct=record
  x,y:integer
end;
cerc=record
  centru:punct;
  raza:real
end;
var p:punct;
    c:cerc;
  
```

**SUBIECTUL al III-lea**

**(30 de puncte)**

- Subprogramul **MaxImp** are doi parametri,  $a$  și  $b$ , prin care primește câte un număr natural ( $2 \leq a < b \leq 400$ ). Subprogramul returnează cel mai mare număr natural din intervalul  $[a, b]$  pentru care produsul divizorilor săi impari pozitivi este strict mai mare decât el însuși sau 0, dacă nu există niciun astfel de număr. Scrieți definiția completă a subprogramului. **Exemplu:** dacă  $a=14$  și  $b=19$ , atunci subprogramul returnează 18 ( $1 \cdot 3 \cdot 9 = 27 > 18$ ). **(10p.)**

- Numim **pătrat** de dimensiune  $m$  al unui tablou bidimensional tabloul obținut din acesta păstrând doar elementele aflate pe primele  $m$  linii și pe primele  $m$  coloane ale sale. Scrieți un program Pascal care citește de la tastatură un număr natural,  $n$  ( $n \in [2, 20]$ ), apoi elementele unui tablou bidimensional cu  $n$  linii și  $n$  coloane, numere naturale din intervalul  $[0, 10^4]$ . Programul determină un pătrat de dimensiune maximă al tabloului citit, cu toate elementele egale, și afișează pe ecran valoarea acestei dimensiuni. **Exemplu:** pentru  $n=5$  și tabloul alăturat, se afișează pe ecran 3. **(10p.)**

```

2 2 2 2 2
2 2 2 2 8
2 2 2 2 2
2 1 2 8 7
3 5 2 1 2
  
```

- Fișierul **bac.txt** conține un șir de cel puțin două și cel mult  $10^6$  numere naturale din intervalul  $[0, 10^3]$ , separate prin câte un spațiu. Șirul are cel puțin un termen par și cel puțin un termen impar. Se cere să se afișeze pe ecran termenii șirului, separați prin câte un spațiu, astfel încât toți cei impari să apară înaintea tuturor celor pari, și atât subșirul format din cei impari, cât și subșirul format din cei pari, să fie în ordine crescătoare, ca în exemplu. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

**Exemplu:** dacă fișierul conține numerele 12 2 3 1 2 5 se afișează pe ecran: 1 3 5 2 2 12

- Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- Scrieți programul Pascal corespunzător algoritmului proiectat. **(8p.)**