

EXAMENUL DE BACALAUREAT – 2007
Proba scrisă la INFORMATICĂ
PROBA E, limbajul Pascal

Varianta 17

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru este de 3 ore.
- ♦ În programele cerute la subiectele II și III, identificatorii utilizați 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).

SUBIECTUL I (40 de puncte)

Pentru fiecare din itemii de la 1 la 8, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 5 puncte.

1. Fiecare element al unei liste liniare simplu înlănțuită alocată dinamic reține în câmpul **adru** adresa elementului următor din listă. Dacă **p** reține adresa primului element, iar lista are cel puțin două elemente, care dintre următoarele secvențe de instrucțiuni șterge al doilea element al listei?
 - a. `q:=p^.adru; p^.adru:=q^.adru; dispose(q)`
 - b. `p^.adru:=p^.adru^.adru; dispose(p^.adru)`
 - c. `q:=p^.adru; dispose(q); p^.adru:=q^.adru`
 - d. `dispose(p^.adru)`
2. Care dintre următoarele secvențe calculează suma elementelor de pe linia **k** a unei matrice **a**, cu **m** linii (numerotate de la 1 la **m**) și **n** coloane (numerotate de la 1 la **n**)?
 - a. `s:=0; for i:=m downto 1 do s:=s+a[k,i]`
 - b. `s:=0; i:=1; while i<=m do begin s:=s+a[i,k]; i:=i+1 end`
 - c. `s:=0; for i:=n downto 1 do s:=s+a[i,k]`
 - d. `s:=0; i:=1; while i<=n do begin s:=s+a[k,i]; i:=i+1 end`
3. Un graf neorientat este eulerian dacă:
 - a. este conex și conține cel puțin un ciclu elementar
 - b. conține un singur ciclu elementar
 - c. este conex și suma elementelor de pe fiecare coloană a matricei de adiacență este număr par
 - d. conține cel puțin un ciclu hamiltonian
4. Știind că funcția **cmmdc**, cu doi parametri întregi **a** și **b**, returnează valoarea **celui mai mare divizor comun** al numerelor **a** și **b**, stabiliți care dintre următoarele variante atribuie variabilei **m** valoarea **celui mai mare divizor comun** al numerelor **a**, **b** și **c**.

<ol style="list-style-type: none"> a. <code>m:=cmmdc(cmmdc(a,b),c)</code> c. <code>m:=cmmdc(a,b,c)</code> 	<ol style="list-style-type: none"> b. <code>m:=cmmdc(a,b)*c</code> d. <code>m:=cmmdc(a,b) div c</code>
---	--
5. Ce valoare trebuie să primească la apel parametrul formal **n**, pentru ca funcția alăturată să returneze valoarea 21?

- a. 7
 - b. 8
 - c. 4
 - d. 6

```
Function ex(n:integer):integer;
begin
  if n=0 then ex:=0 else ex:=n+ex(n-1)
end;
```
6. Într-o listă liniară simplu înlănțuită alocată dinamic fiecare element conține în câmpul **nume** numele și prenumele unei persoane, iar în câmpul **urm** adresa elementului următor. Știind că variabila **x** memorează adresa unui element din listă, cum procedăm pentru a verifica dacă valoarea câmpului **nume** al variabilei **x** este egală cu șirul de caractere „Popescu Ion”?
 - a. `if pos(x^.nume,'Popescu Ion')=1 then`
 - b. `if x='Popescu Ion' then`
 - c. `if x^.nume='Popescu Ion' then`
 - d. `if x.nume='Popescu Ion' then`
7. Care este gradul maxim posibil al unui nod dintr-un arbore cu **n** noduri?
 - a. **n-1**
 - b. **n DIV 2**
 - c. **2**
 - d. **n**

8. Problema generării tuturor numerelor de n cifre, folosind doar cifrele 1, 5 și 7, este echivalentă cu problema:
- generării produsului cartezian a 3 mulțimi cu câte n elemente fiecare
 - generării aranjamentelor de n elemente luate câte 3
 - generării produsului cartezian a n mulțimi cu câte 3 elemente fiecare
 - generării combinațiilor de n elemente luate câte 3

SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat:

- Ce va afișa algoritmul pentru $a=3$ și $b=10$? (5 p.)
- Scrieți algoritmul pseudocod, echivalent cu algoritmul dat, care să folosească un alt tip de structură repetitivă. (5 p.)
- Scrieți programul **Pascal** corespunzător algoritmului. (8 p.)
- Pentru câte perechi de valori (a,b) , cu a și b aparținând intervalului $[1,10]$, rezultatul afișat este egal cu 10? (2 p.)

```

citește a,b {a,b ∈ N}
dacă a<b atunci
    a←a-b
    b←a+b
    a←b-a
    ■
k←0
cât timp a≥b execută
    a←a-b
    k←k+2
    ■
scrie k

```

SUBIECTUL III (30 de puncte)

- Scrieți programul **Pascal** care afișează pe ecran, separate prin virgulă, toate numerele naturale de patru cifre care au prima cifră strict mai mică decât a doua, a doua cifră strict mai mare decât a treia și a treia cifră strict mai mică decât a patra.
Exemplu: 1201,1202,1203,...,8979,8989 (10p.)
- Pentru două numere naturale n și m cu maximum nouă cifre fiecare, citite de la tastatură, numere care au cifrele în ordine crescătoare, se cere să se afișeze pe ecran cel mai mare număr care se poate forma cu toate cifrele numerelor n și m . De exemplu pentru $n=35679$ și $m=123789$ se va tipări 99877653321. Alegeți un algoritm eficient de rezolvare.
 - Descrieți strategia de rezolvare și justificați eficiența algoritmului ales, folosind limbajul natural (5-6 rânduri). (2p.)
 - Scrieți programul **Pascal** corespunzător. (8p.)
- Subprogramul **majuscula** primește prin parametrul c un caracter pe care îl transformă în literă mare dacă este literă mică, în caz contrar lăsându-l nemodificat. Subprogramul returnează valoarea 1 dacă parametrul c este o vocală (a,e,i,o,u,A,E,I,O,U) și 0 altfel.
 - Scrieți doar antetul subprogramului **majuscula**. (2p.)
 - Fișierul **bac.in** conține un rând de text format din maximum 1000 de caractere (litere mari sau mici). Scrieți programul **Pascal** care citește textul din fișierul **bac.in** și îl scrie în fișierul **bac.out** modificat, prin apeluri ale subprogramului **majuscula**, astfel: toate caracterele din text să fie litere mari și după fiecare vocală să apară caracterul semnul exclamării (!).
Exemplu: Fișierul „bac.in” conține: EasdujHIrtoua Fișierul „bac.out” va conține: E!A!SDU!JHI!RTO!U!A! (8p.)