

**CONCURSUL DE OCUPARE A POSTURILOR DIDACTICE/CATEDRELOR DECLARATE
VACANTE/REZERVATE ÎN UNITĂȚILE DE ÎNVĂȚĂMÂNT PREUNIVERSITAR
17 iulie 2019**

**Probă scrisă
INFORMATICĂ ȘI TEHNOLOGIA INFORMAȚIEI**

MODEL

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 4 ore.
- Programele cerute vor fi scrise folosind unul dintre limbajele de programare Pascal, C sau C++, la alegere. Identificatorii utilizați în programe trebuie să corespundă semnificației asociate acestora, eventual în formă prescurtată.

SUBIECTUL I (30 de puncte)

1. Prezentați subprogramele recursive după următorul plan de idei:
- noțiuni specifice (recursivitate directă, recursivitate indirectă);
- mecanismul apelului recursiv pentru două tipuri diferite de subprograme (proceduri/funcții), alese adecvat (descriere și exemplificare din punctul de vedere al evoluției valorilor variabilelor locale/parametrilor);
- un avantaj și un dezavantaj al utilizării subprogramelor recursive;
- un exemplu de utilizare a unui subprogram recursiv în rezolvarea unei probleme concrete (enunț, implementare în limbaj de programare a unei soluții, descriere a soluției).

(15 puncte)

2. Prezentați securitatea și protecția sistemelor de calcul după următorul plan de idei:
- noțiuni preliminare (sistem de calcul, sistem de operare, rețele de calculatoare);
- trei elemente de securitate și protecție (câte două caracteristici pentru fiecare).

(15 puncte)

SUBIECTUL al II-lea (30 de puncte)

1. O listă simplu înlănțuită memorează, în fiecare dintre nodurile sale, câte un număr natural din intervalul $[1, 10^4]$, precum și adresa nodului care îi urmează în listă, dacă acesta există, sau adresa nulă în caz contrar; pentru fiecare nod al listei, memoria este alocată dinamic. Parcurgând lista de la primul către ultimul nod, numerele memorate sunt în **ordine strict crescătoare**.

Subprogramul **caut** are doi parametri:

- **p**, prin care primește adresa primului nod al unei liste simplu înlănțuite de tipul precizat mai sus;
- **x**, prin care primește un număr natural ($x \in [1, 10^4]$).

Subprogramul returnează adresa acelui nod din listă care memorează cel mai mare număr din intervalul $[1, x]$ sau adresa nulă dacă nu există niciun astfel de nod.

Exemplu: pentru lista care conține numerele 3 4 5 9, **p** memorează adresa lui 3; dacă **x=7**, atunci, în urma apelului, subprogramul returnează adresa lui 5, iar dacă **x=2** subprogramul returnează adresa nulă.

Scrieți un program Pascal/C/C++ care citește de la tastatură un număr natural **n** ($n \in [2, 10^4]$) și un șir de **n** numere naturale distincte din intervalul $[1, 10^4]$ și construiește o listă simplu înlănțuită de tipul precizat mai sus, care memorează în nodurile sale numerele din șir. Programul afișează pe ecran, separate prin câte un spațiu, numerele din lista obținută, parcursă de la primul către ultimul nod. Programul cuprinde definiția completă a subprogramului precizat mai sus, precum și apeluri utile ale acestuia.

Exemplu: dacă **n=6** și se citesc numerele 5 2 9 1 6 3

lista obținută memorează, în această ordine, numerele: 1 2 3 5 6 9

(15 puncte)

2. Fișierul `titu.in` conține numere naturale din intervalul $[1, 10^2]$: pe prima linie două numere, m și n , pe a doua linie un șir de m numere, iar pe a treia linie un șir de n numere. Numerele aflate pe aceeași linie a fișierului sunt separate prin câte un spațiu.

Se cere să se afișeze pe ecran numărul de termeni ai celui mai lung subșir comun celor două șiruri aflate în fișier. Utilizați un algoritm eficient din punctul de vedere al timpului de executare.

Exemplu: dacă fișierul are conținutul alăturat, se afișează pe ecran valoarea 3.

```
5 8
4 7 9 8 3
1 4 2 9 7 6 8 2
```

Scrieți programul Pascal/C/C++ corespunzător cerinței și explicați în limbaj natural metoda de rezolvare, justificând eficiența acesteia.

(15 puncte)

SUBIECTUL al III-lea

(30 de puncte)

Se consideră secvențele de mai jos, notate cu **A** și **B**, extrase din programele școlare de liceu pentru disciplinele informatică și tehnologia informației și a comunicațiilor:

A:

Competențe specifice	Conținuturi
3.1. Analizarea enunțului unei probleme și stabilirea pașilor de rezolvare a problemei. 3.2. Reprezentarea algoritmilor în pseudocod. 3.3. Respectarea principiilor programării structurate în procesul de elaborare a algoritmilor.	Reprezentarea algoritmilor. Pseudocod [...] Structuri de bază: <ul style="list-style-type: none">structura liniară

(Programe școlare de INFORMATICĂ, OMECI nr. 5099/09.09.2009)

B:

Competențe specifice	Conținuturi
1.1. Identificarea componentelor hard și soft ale unui calculator personal	<ul style="list-style-type: none">Dispozitive de ieșire

(Programe școlare de TEHNOLOGIA INFORMAȚIEI ȘI A COMUNICAȚIILOR, OMECI nr. 099/09.09.2009)

1. Pentru secvența **A** elaborați un test și baremul de evaluare și de notare corespunzător acestuia, test care să fie utilizat în cadrul unei probe scrise ca instrument de evaluare a competențelor specifice indicate pe baza conținuturilor corespunzătoare. Testul cuprinde enunțurile a cinci itemi, iar baremul de evaluare și de notare distribuie 90 de puncte, acordând 10 puncte din oficiu, și cuprinde, pentru fiecare item, răspunsul așteptat și criteriile de evaluare și de notare analitică a acestuia.

(15 puncte)

2. Pentru secvența **B**, prezentați aspecte ale activității didactice corespunzătoare, în care utilizați **modelarea** ca metodă didactică, având în vedere:

- precizarea a trei caracteristici ale utilizării acestei metodei didactice din perspectiva formării/dezvoltării competențelor specifice indicate pe baza conținuturilor corespunzătoare;

- exemplificarea utilizării metodei, precizând unele elemente ale proiectării didactice: un mijloc de învățământ utilizat, o formă de organizare a clasei, o activitate de învățare și scenariul didactic pentru aceasta, detaliind activitatea profesorului și activitatea elevilor, cu respectarea corectitudinii științifice a informației de specialitate.

(15 puncte)