

**Examenul de bacalaureat național 2020
Proba E. d)
Informatică
Limbajul Pascal**

Testul 9

Filieră teoretică, profil real, specializare științe ale naturii

- 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.

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.

- Variabila x este de tip întreg și memorează un număr nenul. Indicați expresia Pascal cu valoarea `true` dacă și numai dacă 2020 este divizor al lui x .

a. <code>x div (x div 2020)=0</code>	b. <code>x div (x mod 2020)=0</code>
c. <code>x mod (x mod 2020)=0</code>	d. <code>x mod (x div 2020)=0</code>
- Variabilele i și j sunt de tip întreg. Indicați expresia care poate înlocui zona punctată astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran valorile alăturate.

<pre>for i:=0 to 5 do begin for j:=0 to 5 do write('.....', ' '); writeln end;</pre>	<pre>0 0 0 0 0 0 0 2 2 4 4 6 0 2 0 2 0 2 0 4 2 6 4 8 0 4 0 4 0 4 0 6 2 8 4 10</pre>
--	---

a. <code>(i div 2)*j+(j div 2)*i</code>	b. <code>(i mod 2)*j+(j mod 2)*i</code>
c. <code>(i div 2)*i+(j div 2)*j</code>	d. <code>(i mod 2)*i+(j mod 2)*j</code>
- Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea $x=19$ se aplică metoda căutării binare, iar succesiunea de elemente a căror valoare se compară cu x pe parcursul aplicării metodei este 8, 20, 19. Elementele tabloului pot fi:

a. (4, 8, 10, 20, 15, 19, 70)	b. (8, 20, 19, 47, 50, 70)
c. (2, 5, 7, 8, 19, 20, 70)	d. (8, 8, 20, 20, 19, 19)
- O expresie Pascal care are valoarea `true` este:

a. <code>round(20) - 1 = trunc(20)</code>	b. <code>round(20.91) = trunc(20.19)</code>
c. <code>round(20.19) = trunc(20.91)</code>	d. <code>round(20.19) = round(20.91)</code>
- Variabilele x , y , z , w și r sunt de tip întreg, iar r are inițial valoarea 0. Indicați o secvență echivalentă cu cea de mai jos.


```
if x=y then if z=w then r:=1 else r:=2 else r:=3;
```

a. <code>if (x=y) and (z=w) then r:=1 else if (x=y) and (z<>w) then r:=2 else r:=3;</code>	b. <code>if (x=y) or (z=w) then r:=1 else if (x=y) or (z<>w) then r:=2 else r:=3;</code>
c. <code>if (x=y) and (z=w) then r:=1 else if (x=y) and (z<>w) then r:=2 else if (x<>y) and (z<>w) then r:=3;</code>	d. <code>if (x=y) or (z=w) then r:=1 else if (x=y) or (z<>w) then r:=2 else if (x<>y) or (z<>w) then r:=3;</code>

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 întreagă a numărului real c .
- a. Scrieți ce se afișează dacă se citesc, în această ordine, numerele 8 și 3. **(6p.)**
- b. Dacă pentru variabila k se citește valoarea 5, scrieți cea mai mică și cea mai mare valoare care pot fi citite pentru variabila n astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, ultimul număr afișat să fie 10. **(6p.)**
- c. Scrieți programul Pascal corespunzător algoritmului dat. **(10p.)**
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat ultima structură **pentru...execută** cu o structură repetitivă de alt tip. **(6p.)**
2. În secvența alăturată, variabila c este de tip `char`, iar variabilele i și k sunt de tip întreg. Scrieți secvența alăturată, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei k să fie egală cu numărul de litere i citite.
- Exemplu:** dacă se citesc literele
`i m p o n d e r a b i l i t a t e`
variabila k este 3 **(6p.)**
- ```

citește n,k
(numere naturale nenule)
t←1
┌pentru i←1,[n/k] execută
│┌pentru j←1,k execută
││scrie t,' '
││■
││t←t+1
││■
│└pentru i←n%k,1,-1 execută
││scrie t,' '
││■
└
```
3. Tablourile unidimensionale  $A$  și  $B$  au valorile:  $A=(3,5,6,19,20)$  și  $B=(56,15,14,11,1)$ . Scrieți elementele tabloului obținut în urma interclasării lor în ordine crescătoare. **(6p.)**

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Se citește un număr natural,  $n$ , și se cere să se scrie suma cifrelor impare distincte din scrierea acestuia. Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.  
**Exemplu:** dacă  $n=4713835$ , se scrie 16 ( $16=7+1+3+5$ ), iar dacă  $n=48$ , se scrie 0. **(10p.)**
2. Scrieți un program Pascal care citește de la tastatură un număr natural  $n$  ( $n \in [2, 20]$ ) și un șir de  $n$  numere naturale din intervalul  $[1, 10^9]$ , elemente ale unui tablou unidimensional. Programul afișează pe ecran, separate prin câte un spațiu, numerele din șir care sunt strict mai mari decât ultimul număr citit. Dacă nu există astfel de numere, se afișează pe ecran mesajul **nu exista**.  
**Exemplu:** pentru  $n=6$  și tabloul  $(20, 102, 30, 50, 2101, 40)$ , pe ecran se afișează  
102 50 2101 **(10p.)**
3. Numim **k-secvență** într-un șir de numere naturale, o succesiune de termeni aflați pe poziții consecutive în șir, cu proprietatea că sunt divizibili cu numărul natural nenul  $k$ . **Lungimea** secvenței este egală cu numărul de termeni ai săi.  
Fișierul `bac.txt` conține numere naturale din intervalul  $[0, 10^9]$ : pe prima linie un număr nenul  $k$ , iar pe a doua linie un șir de cel mult  $10^6$  numere, separate prin câte un spațiu. Cel puțin un termen din șir este divizibil cu  $k$ . Se cere să se afișeze pe ecran lungimea maximă a unei  $k$ -secvențe din șirul aflat în fișier. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.  
**Exemplu:** dacă fișierul are conținutul alăturat, se afișează 4
- ```

5
2 10 5 20 21 0 10 60 15 3 9 20 20 5 45
```
- a. Scrieți programul Pascal corespunzător algoritmului proiectat. **(8p.)**
- b. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**