Concurs de admitere
Facultatea de Matematică și Informatică
Universitatea de Vest din Timișoara

Sesiunea Septembrie, 2025, Barem Varianta 1

1. Timp total de lucru: 3 ore
2. Toate subiectele sunt obligatorii!
3. Citiți cu atenție informațiile legate de tipurile de întrebări, prezentate mai jos.

Tipuri de întrebări

Pe foaia de concurs veți întâlni trei tipuri de întrebări. Pentru a le recunoaște mai ușor, în partea dreaptă veți regăsi un șablon pentru răspunsuri, cu indicații legate de tipul de întrebare.

Tipul 1 Pentru întrebările de acest tip veți marca pe foaia de concurs un singur răspuns corect. Le veți recunoaște după modelul indicat în partea dreaptă a acestui rând.

MODEL, marcați exact un răspuns:
1 2 3 4

Punctaj pe întrebare: 6p (standard)

Tipul 2 Pentru întrebările de acest tip veți marca pe foaia de concurs fie un singur răspuns corect, fie două răspunsuri corecte. Le veți recunoaște după modelul indicat în partea dreaptă a acestui rând.

MODEL, marcați 1-2 răspunsuri:
1 2 3 4 5

Punctaj maxim pe întrebare: 6p (standard)+1p (bonus pentru răspuns complet)

Tipul 3 Pentru întrebările de acest tip veți marca pe foaia de concurs fie un singur răspuns corect, fie două răspunsuri corecte, fie trei răspunsuri corecte. Le veți recunoaște după modelul indicat în partea dreaptă a acestui rând.

MODEL, marcați 1-3 răspunsuri:
1 2 3 4 5

Punctaj maxim pe întrebare: 6p (standard)+1p (bonus pentru răspuns complet)

Punctaj

Fiecare întrebare valorează 6 puncte. Dacă o întrebare are mai multe variante corecte, atunci cele 6 puncte se vor împărți egal la numărul de variante corecte. Dacă răspunsul oferit identifică toate variantele corecte de răspuns, la punctajul întrebării se adaugă un bonus pentru răspuns complet de 1 punct. Orice răspuns greșit anulează întregul punctaj al întrebării respective, indiferent de numărul de răspunsuri corecte.
La punctajul obținut se adaugă 10 puncte, pentru start.
Pentru nota 5 sunt necesare 52 de puncte, inclusiv punctele de start, pentru nota 10 sunt necesare cel puțin 130 de puncte, inclusiv punctele de start.

Terminologie și convenții

Tipuri de date

$$\begin{aligned} \mathit{int} &- \text{număr întreg} \\ \mathit{real} &- \text{număr real} \\ \mathit{boolean} &- \text{valoare logică} \end{aligned}$$

Operații și simboluri

$$\begin{aligned} a \ \textbf{DIV}\ b &- \text{calculează câtul împărțirii întregi (cu rest) a lui } a \text{ la } b \\ a \ \textbf{MOD}\ b &- \text{calculează restul împărțirii întregi a lui } a \text{ la } b \\ a \ \textbf{AND}\ b &- \text{conjuncția logică} \\ a \ \textbf{OR}\ b &- \text{disjuncția logică} \\ \textbf{NOT}\ a &- \text{negarea logică} \\ a = 2 &- \text{operația de atribuire a valorii 2 pentru variabila } a \\ a == 2 &- \text{verificarea egalității variabilei } a \text{ cu valoarea 2} \\ a \neq 2 &- \text{verificarea inegalității variabilei } a \text{ cu valoarea 2} \end{aligned}$$

Partea I

Problema 1: Factorial sau nu?

Se consideră funcţiile F1 şi F2 definite în modul ilustrat mai jos pentru un întreg n:

```
funcția F1(n)
    dacă n == 0 atunci
        întoarce 1
    altfel
        întoarce n * F1(n − 1)
```

```
funcția F2(n)
    dacă n != 1 atunci
        întoarce n * F2(n − 1)
    altfel
        întoarce 0
```

Care din funcţiile F1 sau F2 returnează 120 când este apelată cu parametrul de intrare $n = 5$?

Ciornă, marcați exact un răspuns: 1 2 3 4

① atât F1 cât şi F2 ② numai F1 ✔ ③ numai F2 ④ nici F1, nici F2

Explicații

Se observă că, dacă $n \geq 0$ atunci $F1(n)$ returnează valoarea lui $n!$, iar $F2(n)$ returnează 0. Întrucât $120 = 5!$, rezultă că răspunsul corect este "numai F1".

Problema 2: Numere

Câte numere de 5 cifre distincte se pot forma folosind doar cifrele 1, 2, 3, 4, 5, 6, 7, astfel încât numerele generate să fie divizibile cu 4?

① 600 ✔ ② 480 ③ 560 ④ 640

Explicații

Principiul de divizibilitate cu 4 spune că ultimele două cifre ale numărului să fie divizibile cu 4. Astfel, se evaluează ultimele 2 cifre: 12, 16, 24, 32, 36, 52, 56, 64, 72 și 76. Pentru celelalte 3 cifre, avem $5 \times 4 \times 3$ combinații, total $60 \times 10 = 600$.

Problema 3: Din întuneric

Sorin lucrează la hotelul California și are grijă de un culoar cu 10 de becuri, dispuse la distanță egală, fiecare cu întrerupător propriu. Pentru că este nevăzător, el face mai multe ture pentru a se asigura că unele becuri sunt aprinse pentru oaspeți. În prima tură va comuta fiecare întrerupător: dacă e deschis acesta va fi închis, și invers. În a doua tură, va comuta întrerupătoarele corespunzătoare multiplilor de 2, în a treia pe cele corespunzătoare multiplilor de 3, *ș.a.m.d.*.

Câte becuri vor fi aprinse după 5 ture, având în vedere că inițial toate becurile erau stinse?

① exact jumătate: 5 becuri
② doar 4 becuri
③ 6 becuri ✔
④ multă lumină: 7 becuri

Explicații

00000 00000 -> stare inițială becuri
11111 11111 -> situație după tura 1
10101 01010 -> situație după tura 2
10001 11000 -> situație după tura 3
10011 11100 -> situație după tura 4
10010 11101 -> situație după tura 5: 6 becuri aprinse

Problema 4: Identificarea lui C

Pe o insulă se găsesc două grupări de persoane: Vedetele și Războinicii. În timp ce Vedetele spun doar adevărul, Războinicii sunt permanent mincinoși. Aflat în căutarea lor, exploratorul Explorer întâlnește un grup de trei persoane, A, B și C, despre care află următoarele:

- A spune despre B că este din gruparea Războinicilor.
- B spune că atât A cât și C sunt din aceeași grupare.

Din ce grupare face parte C?

Ciornă, marcați exact un răspuns:
1 2 3 4

① C este un Războinic. ✔
② C este o Vedetă.
③ Nu putem determina.
④ C ar putea fi Explorator.

Explicații

Raționamentul lui E este următorul:
Dacă A spune adevărul, atunci B este un Războinic, prin urmare este mincinos, ceea ce înseamnă că A și C nu sunt din același grup. Prin urmare, C este tot Războinic;
Dacă A este un Războinic, atunci afirmația despre B este o minciună, deci B este un Vedetă. Prin urmare, a doua afirmație este adevărată, iar C este Războinic.
Așadar, concluziile la care se poate ajunge sunt următoarele: A și B nu fac parte din același grup; C este Războinic.

Problema 5: Cut the rod

Împăratul Roșu trebuie să construiască o nouă aripă a palatului său. Pentru aceasta, se înțelege cu echipa de *meșteri mari, calfe și zidari*, ca la sfârşitul fiecărei luni k să existe într-un cufăr o cantitate de aur cu greutatea $k \times G$, iar echipa va lua întregul cufăr odată cu finalizarea construcției.

Deoarece întreaga muncă va dura 11 luni, iar Împăratul Roșu are pentru aceasta o bară destul de lungă de aur, cu o greutate egală cu $23 \times G$, acesta își pune problema tăierii acesteia cu un număr minim de tăieturi, astfel încât să poată realiza plățile lunare, după ce bijutierul l-a asigurat că poate decupa orice bucată de greutate $k \times G$ folosind o singură tăietură.

Ciornă, marcați exact un răspuns:
1 2 3 4

① 10 tăieturi. ② 7 tăieturi. ③ 4 tăieturi. ✔ ④ 3 tăieturi.

Explicații

Luna 1: k=1, 1 tăietură
Luna 2: k=2, 1 tăietură (se scoate bucata k=1 și se pune bucata cu k=2)
Luna 3: k=1+2 (se pun ambele bucăți), 0 tăieturi
Luna 4: k=4 (se taie o bucata de 4 și se scot celelalte 2 din cufăr), 1 tăietură
Luna 5: k=4+1 (bucata de 4 și bucata de 1), 0 tăieturi
Luna 6: k=4+2 (bucata de 4 și bucata de 2), 0 tăieturi
Luna 7: k=4+2+1 (toate cele 3 bucăți), 0 tăieturi
Luna 8: k=4+4 (se taie o bucata de 4 și se pune cu cealaltă bucata de 4), 1 tăietură
Luna 9: k=4+4+1 (doua bucăți de 4 și una de 1), 0 tăieturi
Luna 10: k=4+4+2 (doua bucăți de 4 și una de 2), 0 tăieturi
Luna 11: k=4+4+2+1 (toate cele 4 bucăți tăiate), 0 tăieturi

Problema 6: Bile

Într-o cutie se află zece bile albe și zece bile negre. Care este numărul minim de bile care ar trebui extrase din cutie, pe nevăzute, astfel încât să avem cel puțin o pereche alb-negru?

Ciornă, marcați 1-2 răspunsuri: ① ② ③ ④ ⑤

① 2 ② 10 ③ 11 ✔ ④ 19 ⑤ toate

Explicații

În cel mai rău caz se pot extrage consecutiv 10 bile de aceeași culoare. Deci, abia a unsprezecea bilă ne va garanta obținerea unei perechi alb-negru.

Problema 7: O procedură

Se consideră funcția RunMe, de mai jos

```
funcția RunMe(k)
    i, rezultat ← 0
    pentru i ← 0 to k execută
        dacă i MOD 3 = 1 atunci
            rezultat ← rezultat + i
        altfel
            rezultat ← rezultat + 1
    întoarce rezultat
```

Precizați care dintre următoarele afirmații sunt adevărate:

Ciornă, marcați 1-2 răspunsuri: ① ② ③ ④ ⑤

① Pentru apelul RunMe(5) funcția va returna 8

② Pentru apelul RunMe(0) funcția va returna 11

③ Pentru apelul RunMe(5) funcția va returna 5

④ Pentru apelul RunMe(4) funcția va returna 8 ✔

⑤ Pentru apelul RunMe(9) funcția va returna 18

Explicații

Răspunsurile așteptate sunt:
Pentru apelul RunMe(4) funcția va returna 8

Partea a II-a

Problema 8: O recurență cu memoizare

Se dă un tablou `a[1..10000]` inițializat cu valoarea `-1`. Considerăm următoarea secvență de cod, unde $x_n = a[n]$:

$a[1] = 1$
funcția REC1(n)
  dacă $a[n-1] = -1$ **atunci**
    REC1($n-1$)
  $v = a[n-1]$
  $a[n] = \dfrac{v}{3v+5}$
  întoarce $a[n]$

Precizați care dintre afirmațiile de mai jos sunt adevărate:

① Termenul x_n are formula $x_n = \dfrac{1}{\frac{7}{20} \cdot 5^n - \frac{3}{4}}$. ✔

② Relația recurentă care rezultă din programul de mai sus este $x_{n+1} = \dfrac{x_n}{3x_n + 5}$, $x_1 = 1$. ✔

③ Termenul x_n are formula $x_n = \dfrac{1}{\frac{3}{5} \cdot 5^n - \frac{3}{4}}$.

④ Relația recurentă care rezultă din programul de mai sus este $x_{n+1} = \dfrac{x_n}{3x_n + 5}$, $x_0 = 1$.

⑤ Secvența de cod va genera aceeași valoare dacă se înlocuiește ultima linie cu `return rec1(n-1) / (3*rec1(n-1) + 5)`.

Explicații

Din cod rezultă imediat recurența $x_{n+1} = \dfrac{x_n}{3x_n+5}$ cu condiția inițială $x_1 = 1$ (afirmația **(2)** este adevărată).

Punând $y_n = \dfrac{1}{x_n}$, obținem recurența liniară $y_{n+1} = 5y_n + 3$, cu $y_1 = 1$. Soluția este $y_n = \dfrac{7}{4} \cdot 5^{n-1} - \dfrac{3}{4}$, deci

$$x_n = \frac{1}{\left(\frac{7}{4}\right) 5^{n-1} - \frac{3}{4}} = \frac{1}{\left(\frac{7}{20}\right) 5^{n} - \frac{3}{4}},$$

care corespunde afirmației **(1)**. Celelalte afirmații sunt false.

Problema 9: Matrice Toeplitz

Se dă o matrice $\mathtt{mat}[1..n,\ 1..m]$ și funcția FCT de mai jos, care verifică dacă toate diagonalele de la stânga-sus la dreapta-jos au aceeași valoare (proprietate de tip *Toeplitz*):

```
funcția FCT(mat)
    pentru r = 2 to n execută
        pentru c = 2 to m execută
            dacă mat[r][c] ≠ mat[r − 1][c − 1] atunci
                întoarce False
    întoarce True
```

Precizați care dintre următoarele afirmații sunt adevărate:

① Pentru matricea $[[1,2,3],[4,1,2],[5,4,1]]$ funcția întoarce **True**. ✔

② Pentru matricea $[[1,2,3],[4,1,2],[5,4,1],[6,5,4]]$ funcția întoarce **False**.

③ Pentru matricea $[[1,2,3],[4,2,3],[5,4,2]]$ funcția întoarce **False**. ✔

④ Dacă se modifică condiția din `if` cu $\text{mat}[i][j] \neq \text{mat}[i-1][j+1]$, programul verifică corect fiecare *antidiagonală* (de la dreapta-sus la stânga-jos).

⑤ Pentru ca programul de mai sus să verifice că toate diagonalele (de la stânga-sus la dreapta-jos) conțin o valoare constantă, matricea trebuie să fie pătratică.

Explicații

(1) este adevărată: fiecare diagonală principală are elemente egale (matrice Toeplitz).
(2) este falsă: și această matrice are diagonalele principale constante, deci funcția întoarce **True**.
(3) este adevărată: de pildă $\text{mat}[2][2] = 2 \neq \text{mat}[1][1] = 1$, deci funcția întoarce **False**.
(4) este falsă *în forma dată*: pentru a verifica antidiagonalele trebuie ajustate limitele buclei (de ex. $c = 1$ la $m - 1$); altfel accesăm $\text{mat}[i-1][j+1]$ în afara matricii când $j = m$.
(5) este falsă: proprietatea Toeplitz are sens și pentru matrici dreptunghiulare; nu e necesar ca matricea să fie pătratică.

Problema 10: Numere pătrate perfecte

Se consideră următoarele funcții care testează o proprietate a unui număr întreg n:

```
funcția FCT1(n)
    dacă n < 0 atunci
        întoarce False
    i = 1
    cât timp n > 0 execută
        n = n − i
        i = i + 2
    întoarce (n == 0)

funcția FCT2(n)
    s = 0
    i = 1
    cât timp s < n execută
        s = s + i
        i = i + 2
    întoarce (s == n)

funcția FCT3(n)
    întoarce FCT3_AUX(n, 1)

funcția FCT4(n)
    dacă n < 0 atunci
        întoarce False
    întoarce FCT4_AUX(0, n, n)

funcția FCT3_AUX(n, i)
    dacă n < 0 atunci
        întoarce False
    dacă n == 0 atunci
        întoarce True
    întoarce FCT3_AUX(n − i, i + 2)

funcția FCT4_AUX(st, dr, n)
    dacă st > dr atunci
        întoarce False
    mid = (st + dr) DIV 2
    sq = mid ∗ mid
    dacă sq == n atunci
        întoarce True
    altfel dacă sq < n atunci
        întoarce FCT4_AUX(mid + 1, dr, n)
    altfel
        întoarce FCT4_AUX(st, mid − 1, n)
```

Selectați toate variantele adevărate:

① Toate cele patru funcții verifică aceeași proprietate a lui n. ✔

② Doar funcțiile FCT1 și FCT2 verifică aceeași proprietate a lui n.

③ Doar funcțiile FCT1 și FCT4 verifică aceeași proprietate a lui n.

4 Doar funcțiile FCT1 și FCT3 verifică aceeași proprietate a lui n.

5 În intervalul $[100, 250]$ sunt 6 numere pentru care FCT1 întoarce valoarea **True**. ✔

Explicații

Toate cele patru funcții testează dacă n este **pătrat perfect**.
FCT1/FCT2/FCT3 se bazează pe faptul că suma primelor k numere impare este k^2, iar FCT4 face căutare binară pe pătrate (mid^2).
În intervalul $[100, 250]$ pătratele perfecte sunt: $100 = 10^2, 121 = 11^2, 144 = 12^2, 169 = 13^2, 196 = 14^2, 225 = 15^2$ (șase valori). De aceea sunt adevărate afirmațiile **(1)** și **(5)**.

Problema 11: Misterul vârstei medii

Alex a întârziat la una dintre activitățile dintr-un grup de studiu, dar odată cu sosirea lui, vârsta medie a celor aflați în grupul de lucru a crescut cu exact 4 ani. Andreea, sora lui geamănă, a întârziat și mai mult. Și, lucru interesant, odată cu sosirea ei, vârsta medie a celor aflați în sală a crescut cu exact 3 ani.

Câte persoane se aflau în sală, înaintea sosirii lui Alex?

Ciornă, marcați 1-2 răspunsuri: 1 2 3 4 5

1 5   2 6 ✔   3 7   4 8   5 9

Explicații

Fie:
S – suma inițială a vârstelor,
a – vârsta lui Alex și a Andreei,
n – numărul inițial de persoane.
Prima condiție conduce la $\frac{S+a}{n+1} = \frac{S}{n} + 4$, deci $S = a \cdot n - 4 \cdot n \cdot (n+1)$
A doua condiție conduce la $\frac{S+2a}{n+2} = \frac{S+a}{n+1} + 3$, deci $S = a \cdot n - 3 \cdot (n+1) \cdot (n+2)$
Din egalarea relațiilor de mai sus rezultă $4 \cdot n \cdot (n+1) = 3 \cdot (n+1) \cdot (n+2)$ de unde rezultă $n = 6$

Problema 12: Divizibil

$$A = \begin{pmatrix} 15 & 40 & 51 & 61 \\ 44 & 12 & 33 & 71 \\ 2 & 33 & 5 & 54 \\ 12 & 32 & 15 & 45 \end{pmatrix}$$

Fie dată funcția de mai jos și matricea alăturată, A.

```
funcția RunIt(n, m, A)
    mm = 0
    pentru j = 1, m execută
        x = 0
        pentru i = 1, n execută
            dacă A[i][j] MOD 3 = 0 atunci
                x = x + 1
        dacă x > mm atunci
            mm = x
    întoarce mm
```

Care din următoarele variante sunt adevărate?

① $RunIt(3,3,A) \rightarrow 2$ ✔

② $RunIt(2,4,A) \rightarrow 2$ ✔

③ $RunIt(3,3,A) \rightarrow 1$

④ $RunIt(4,4,A) \rightarrow 3$ ✔

⑤ $RunIt(4,4,A) \rightarrow 2$

Ciornă, marcați 1-3 răspunsuri:
1 2 3 4 5

Explicații

Algoritmul va calcula numărul de elemente divizibile cu 3 în fiecare coloană din matricea A, returnând valoarea maximă. Parametrii n și m vor restrânge numărul de linii, respectiv numărul de coloane parcurse.

Problema 13: Sub diagonală

Se consideră o matrice pătratică A de ordin n, și se presupune că elementele aflate sub diagonala principală sunt stocate într-un tablou prin parcurgerea linie cu linie a matricii. Presupunând că indicii tabloului și cei ai matricii pornesc de la 1, care din următoarele afirmații sunt adevărate?

① ✔ $a_{3,2}$ va fi stocat pe poziția 3 în tablou

② $a_{3,1}$ va fi stocat pe poziția 3 în tablou

③ ✔ $a_{i,j}$ va fi stocat pe poziția $\frac{(i-2)(i-1)}{2} + j$ în tablou

④ $a_{i,j}$ va fi stocat pe poziția $\frac{(i)(i+1)}{2} + j$ în tablou

⑤ $a_{3,1}$ va fi stocat pe poziția 4 în tablou

Ciornă, marcați 1-2 răspunsuri:
1 2 3 4 5

Explicații

Primul element de pe diagonala principală este $a_{1,1}$ în acest caz nu se adaugă nici un element în vector.
Al doilea element pe diagonala principală este $a_{2,2}$, în acest caz se adaugă un element în vector.
Al treilea element pe diagonala principală este $a_{3,3}$, în acest caz se adaugă două elemente în vector, deci vectorul conține $1+2$ elemente.
Al i-lea element de pe diagonala principală este $a_{i,i}$, în acest caz se adaugă $i-1$ elemente în vector, deci vectorul conține $1+2+\ldots+(i-1) = \frac{(i-1)i}{2}$ elemente.
În cazul elementului $a_{i,j}$ avem în vector deja elementele până elementul $a_{i-1,i-1}$ de pe diagonală adică $\frac{(i-2)(i-1)}{2}$ la care se adaugă elementele până la poziția j.

Problema 14: Împărțire frățească

Se consideră algoritmul RunMe(n) unde **n** este număr întreg.

```
funcția RunMe(n)
    i = 1, a = 0, k = 0,
```

cât timp $n \neq 0$ **execută**
  $a = a + (n$ **MOD** $2) * i$
  dacă a **MOD** $2 \neq 0$ **atunci**
    $k = k + 1$
  $n = n$ **DIV** 2
  $i = i * 10$
afișează a

Precizați care dintre următoarele afirmații sunt adevărate:

(1) Pentru apelul RunMe(6), algoritmul va afișa 110 ✔

(2) Pentru apelul RunMe(117), algoritmul va afișa 9

(3) Variabila k stochează numărul de cifre impare din n

(4) Pentru apelul RunMe(2), algoritmul va afișa 10 ✔

(5) Pentru apelul RunMe(5), k va avea valoarea finală 3 ✔

Ciornă, marcați 1-3 răspunsuri:
(1) (2) (3) (4) (5)

Explicații

Răspunsurile așteptate sunt:
Pentru apelul RunMe(6), algoritmul va afișa 110
Pentru apelul RunMe(2), algoritmul va afișa 10
Pentru apelul RunMe(5), k va avea valoarea finală 3

Partea a III-a

Problema 15: Calcul

Se consideră algoritmul RunMe(n), unde n este un număr natural nenul. Algoritmul este definit astfel:

funcția RunMe(n)
  dacă $n = 1$ **atunci**
    întoarce 1
  întoarce RunMe(n **DIV** 2)

Precizați care dintre următoarele afirmații sunt adevărate:

(1) Algoritmul calculează $\log_2 n$.

(2) Algoritmul returnează valoarea n^2.

(3) Algoritmul returnează 1 pentru orice valoare a lui n ✔

(4) Algoritmul are mereu timp finit de execuție, cât timp facem împărțirea întreagă la 2 ✔

(5) Algoritmul rulează la infinit pentru numere pare.

Ciornă, marcați 1-3 răspunsuri:
(1) (2) (3) (4) (5)

Explicații

Răspunsurile așteptate sunt:
Algoritmul returnează 1 pentru orice valoare a lui n
Algoritmul are mereu timp finit de execuție, cât timp facem împărțirea întreagă la 2

Problema 16: Un mister

Se consideră algoritmul RunMe(n) unde n este un număr natural cu $1 \leq n \leq 1000$.

funcția RunMe(n)
  dacă $n < 2$ **atunci**
    întoarce n
  $a \leftarrow 0$
  $b \leftarrow 1$
  $c \leftarrow 0$
  pentru i ←2 to n **execută**
    $c \leftarrow a + b$
    $a \leftarrow b$
    $b \leftarrow c$
  întoarce c

Precizați care dintre următoarele afirmații sunt adevărate:

(1) Pentru apelul RunMe(5), algoritmul returnează 5 ✔
(2) Pentru apelul RunMe(8), algoritmul returnează 21 ✔
(3) Pentru apelul RunMe(10), algoritmul returnează 55 ✔
(4) Pentru apelul RunMe(15), algoritmul returnează 600
(5) Pentru apelul RunMe(20), algoritmul returnează 2 765

Ciornă, marcați 1-3 răspunsuri:
(1) (2) (3) (4) (5)

Explicații

Algoritmul este un șir de valori, de tip Fibonacci: [0, 1, 2, 3, 5, ..., s(n-1) + s(n-2)]
Răspunsurile așteptate sunt:
Pentru apelul RunMe(5), algoritmul returnează 5
Pentru apelul RunMe(8), algoritmul returnează 21
Pentru apelul RunMe(10), algoritmul returnează 55

Problema 17: Apel ciudat

Se consideră algoritmul RunMe(n), unde n este un număr natural.
Algoritmul este definit astfel:

funcția RunMe(n)
  dacă $n < 10$ **atunci**

```
    întoarce n
  întoarce RunMe(n DIV 10) * (n MOD 10)
```

Precizați care dintre următoarele afirmații sunt adevărate:

① Algoritmul calculează produsul cifrelor numărului n. ✔

② Algoritmul returnează numărul obținut prin inversarea ordinii cifrelor lui n.

③ Algoritmul returnează valoarea n, nemodificată.

④ Algoritmul returnează 1 pentru orice valoare a lui n.

⑤ Algoritmul calculează suma cifrelor numărului n.

Ciornă, marcați 1-3 răspunsuri:
① ② ③ ④ ⑤

Explicații

Răspunsul așteptat este:
Algoritmul calculează produsul cifrelor numărului n.

Problema 18: Înmulțire repetată

Se consideră algoritmul RunMe(n) unde $0 \leq n \leq 1000$.

```
funcția RunMe(n)
  rezultat ← 0
  pentru i ←1 to n execută
    pentru j ←1 to i execută
      rezultat ← rezultat + i * j
  întoarce rezultat
```

Precizați care dintre următoarele afirmații sunt adevărate:

① Pentru apelul RunMe(3), algoritmul returnează valoarea 25 ✔

② Pentru apelul RunMe(5), algoritmul returnează valoarea 104

③ Pentru orice apel RunMe(7), algoritmul returnează valoarea 262

④ Pentru $n = 0$, algoritmul returnează valoarea 0 ✔

⑤ Pentru orice apel, algoritmul returnează valoarea $n * (n + 1)/2$

Ciornă, marcați 1-3 răspunsuri:
① ② ③ ④ ⑤

Explicații

Răspunsurile așteptate sunt:
Pentru apelul RunMe(3), algoritmul returnează valoarea 25
Pentru $n = 0$, algoritmul returnează valoarea 0

Problema 19: Happy counter

Se consideră algoritmul RunMe(n), unde n este un număr natural.
Algoritmul este definit astfel:

```
funcția RunMe(n)
    rezultat ← 0
    pentru i ←1 to n execută
        dacă i MOD 3 = 0 atunci
            rezultat ← rezultat + i
        dacă i MOD 5 = 0 atunci
            rezultat ← rezultat + i
    întoarce rezultat
```

Precizați care dintre următoarele afirmații sunt adevărate:

Ciornă, marcați 1-3 răspunsuri: (1) (2) (3) (4) (5)

(1) Pentru apelul RunMe(10), algoritmul returnează 33 ✓

(2) Pentru apelul RunMe(15), algoritmul returnează 75 ✓

(3) Pentru apelul RunMe(30), algoritmul returnează 170

(4) Algoritmul calculează suma tuturor numerelor de la 1 la n, care sunt divizibile cu 3 și 5.

(5) Algoritmul calculează suma tuturor numerelor de la 1 la n, care sunt divizibile cu 3 sau 5 sau 15.

Explicații

Răspunsurile așteptate sunt:
Pentru apelul RunMe(10), algoritmul returnează 33
Pentru apelul RunMe(15), algoritmul returnează 75

Problema 20: O traversare în adâncime

Se consideră labirintul ilustrat mai jos, în care celulele negre reprezintă obstacole.

	1	2		3
4		G	5	6
S	7		8	
9	10	11	12	13
		14	15	

Se pune problema găsirii unui drum de la celula de start S la celula G cu ajutorul metodei de parcurgere în adâncime (Depth First). Pe durata parcurgerii în adâncime se aplică următoarele reguli:

1. Vizitarea celulelor se face în ordinea: sus, dreapta, jos, stânga. Nu este posibilă mutarea pe diagonală.
2. Nici o celulă nu se poate vizita de 2 ori.

De exemplu, căutarea în adâncime a unui drum pornind de la S începe cu S, 4, 7, 10, 11 în această ordine. Fie σ secvenţa de celule de pe drumul găsit de căutarea în adâncime, care începe cu S şi se termină cu G.

Care din afirmaţiile următoare sunt adevărate? Marcaţi toate variantele corecte.

Ciornă, marcați 1-3 răspunsuri: (1) (2) (3) (4) (5)

(1) σ are 8 celule.

(2) σ are 11 celule.✓

3 13 este în σ.

4 3 este în σ.✔

5 15 este în σ.

Explicații

Întrucât $\sigma = [S, 4, 7, 10, 11, 12, 8, 5, 6, 3, G]$, răspunsurile corecte sunt 2 și 4.