

Simulare Examen Admitere UBB 2024 – 18-19 Mai 2024
Proba scrisa la Informatica

NOTA IMPORTANTA

In lipsa altor precizări:

- Presupuneti ca toate operatiile aritmetice se efectuează pe tipuri de date nelimitate (nu exista overflow / underflow).
- Numerotarea indicilor tuturor şirurilor / vectorilor începe de la 1.
- Toate restricțiile se refera la valorile parametrilor actuali la momentul apelului inițial.

1. Precizati care dintre urmatoarele expresii verifica corect daca numarul **N** ($1 \leq N \leq 10^9$) este divizibil cu 24:
 - A. $(N \bmod 6) = 0 \text{ AND } (N \bmod 4) = 0$
 - B. $(N \bmod 4 = 0) \text{ AND } ((N \text{ DIV } 4) \bmod 2 = 0) \text{ AND } (N \bmod 6) = 0$
 - C. $(N \bmod 8) = 0 \text{ AND } (N \bmod 3) = 0$
 - D. $((N \text{ DIV } 4) \bmod 2 = 0) \text{ AND } (N \bmod 6 = 0)$
2. Se considera subalgoritmul *ce_face* care primeste ca si parametru unic de intrare un numar natural nenul **n** ($1 \leq n \leq 10^9$):

Subalgorithm *ce_face*(n):

$C \leftarrow [0] * (n + 1)$

$C[1] \leftarrow 1$

$C[0] \leftarrow 1$

For $i \leftarrow 2, \sqrt{n}$ **execute**

If $C[i] = 0$ **then**

For $j \leftarrow i, n \text{ DIV } i$ **execute**

$C[i * j] \leftarrow 1$

EndFor

EndIf

EndFor

return $C[n] = 0$

EndSubalgorithm

Precizati care dintre urmatoarele afirmatii referitoare la subalgoritmul *ce_face* sunt adevarate:

- A. Subalgoritmul verifica daca numarul n este numar par.
- B. Subalgoritmul verifica daca numarul n este prim.
- C. Subalgoritmul verifica primalitatea numarului n printr-un algoritm de complexitate timp minima.
- D. Subalgoritmul returneaza 1 pentru orice numar natural impar transmis ca si parametru.

3. Se considera subalgoritmul f definit mai jos care primește ca si parametrii de intrare un sir a cu n elemente numere naturale nenule si un numar natural nenul k ($1 \leq n \leq 10^6$, $1 \leq a[i] \leq 10^9$, $1 \leq i \leq n$, $1 \leq k \leq n$):

Subalgorithm $f(a, n, k)$:

$\text{maxi} \leftarrow 1$

For $i \leftarrow 1, n - k + 1$ **execute**

$\text{val} \leftarrow a[i]$

For $j \leftarrow i + 1, i + k - 1$ **execute**

While $a[j] \neq 0$ **execute**

$r \leftarrow \text{val} \bmod a[j]$

$\text{val} \leftarrow a[j]$

$a[j] \leftarrow r$

EndWhile

EndFor

If $\text{val} > \text{maxi}$ **then**

$\text{maxi} \leftarrow \text{val}$

EndIf

EndFor

return maxi

EndSubalgorithm

Precizati care dintre urmatoarele afirmatii referitoare la subalgoritmul f definit mai sus sunt adevarate:

- A. Subalgoritmul determina secventa de lungime maxim k cu cel mai mare CMMDC al elementelor din sir.
 - B. Subalgoritmul determina cel mai mare CMMDC al unei secvente de k elemente din sir.
 - C. Subalgoritmul nu calculeaza corect CMMDC-ul fiecarei secvente de lungime k din sir.
 - D. Pentru $k = 1$ se returneaza valoarea maxima din sir.
4. Se consider subalgoritmul suma definit mai jos care primește ca si parametrii de intrare un sir a cu n elemente numere naturale nenule, ordonat crescator si o variabila val , de asemenea numar natural nenul ($1 \leq n \leq 10^6$, $1 \leq a[i] \leq 10^9$, $1 \leq i \leq n$, $1 \leq k \leq 10^6$):

Subalgorithm $\text{suma}(a, n)$:

$\text{cnt} \leftarrow 0$

$\text{st} \leftarrow 1$

$\text{dr} \leftarrow n$

While $\text{st} < \text{dr}$ **execute**

If $a[\text{st}] + a[\text{dr}] = \text{val}$ **then**

$\text{cnt} \leftarrow \text{cnt} + 1$

$\text{st} \leftarrow \text{st} + 1$

$\text{dr} \leftarrow \text{dr} - 1$

EndIf

If $a[\text{st}] + a[\text{dr}] > \text{val}$ **then**

```

        dr ← dr - 1
    EndIf
    If a[st] + a[dr] < val then
        st ← st + 1
    EndIf
EndWhile
Return cnt
EndSubalgorithm

```

Precizati care dintre urmatoarele afirmatii sunt adevarate referitoare la subalgoritmul *suma*(a, n):

- A. Subalgoritmul determine numarul de perechi de elemente, egal departate de capetele sirului, al carei suma este egala cu val.
 - B. Subalgoritmul determina cate perechi de elemente exista in sir pentru care suma lor sa fie egala cu val.
 - C. Pentru un n par, se returneaza mereu o valoare para.
 - D. Niciuna dintre afirmatiile A, B sau C nu este corecta.
5. Câte noduri are un arbore cu rădăcină în care numărul de muchii este egal cu numărul de muchii ale unui graf neorientat complet cu n noduri?
- A. $\frac{n^2 - n + 2}{2}$
 - B. $\frac{n * (n + 1)}{2}$
 - C. $\frac{n * (n + 1)}{2} + 1$
 - D. $\frac{n * (n + 1)}{2} - 1$
6. Se considera subalgoritmul *afis* care primeste ca si parametrii de intrare o matrice patratica a cu n linii si n coloane si numarul natural nenul n ($1 \leq n \leq 10^3$, $1 \leq a[i][j] \leq 10^9$, $1 \leq i, j \leq n$):

```

Subalgorithm afis(a, n):
    cnt ← 0
    i ← 1
    j ← 1
    While cnt < n * n execute
        While j ≤ n AND cnt < n * n execute
            Write a[i][j], ' '
            j ← j + 1
            cnt ← cnt + 1
        EndWhile
        While i ≤ n AND cnt < n * n execute
            Write a[i][j-1], ' '
            i ← i + 1
            cnt ← cnt + 1
        EndWhile
        i ← i - 1
        j ← j - 1
        While j ≥ 1 AND cnt < n * n execute

```

```

        Write a[i][j], ' '
        j ← j - 1
        cnt ← cnt + 1
    EndWhile
    While i ≥ 1 AND cnt < n * n execute
        Write a[i][j+1], ' '
        i ← i - 1
        cnt ← cnt + 1
    EndWhile
    i ← i + 1
    j ← j + 1
EndWhile
EndSubalgorithm

```

Stiind ca subalgoritmul se apeleaza pentru o matrice generata dupa formula $a[i][j] = 2 * i + j$ si $n = 4$, ce se afiseaza?

- A. 3, 4, 5, 6, 8, 10, 12, 11, 10, 9, 7, 5, 6, 7, 9, 8
- B. 3, 4, 5, 6, 8, 10, 12, 11, 10, 9, 7, 5, 3, 4, 5, 6
- C. 3, 4, 5, 6, 6, 8, 10, 12, 12, 11, 10, 9, 9, 7, 5, 3
- D. 3, 4, 5, 6, 5, 6, 7, 8, 7, 8, 9, 10, 9, 10, 11, 12

7. Se considera subalgoritmul *ce_face* care primeste ca si parametrii o matrice **a** cu **n** linii si **n** coloane in care elementele de pe fiecare linie luate una dupa cealalta reprezinta elementele unui vector ordonat crescator, parametru **n**, numar natural nenul si o valoare **val**, de asemenea numar natural ($1 \leq n \leq 10^3$, $1 \leq a[i][j] \leq 10^9$, $1 \leq i, j \leq n$, $a[1][1] \leq a[1][2] \leq \dots \leq a[1][n] \leq a[2][1] \leq a[2][2] \leq \dots \leq a[2][n] \leq a[3][1] \leq \dots$)

```

Subalgorithm ce_face(a, n):
    st ← 1
    dr ← n
    While st ≤ dr execute
        mij ← (st + dr) DIV 2
        If a[mij][n - mij + 1] = val then
            Return 1
        EndIf
        If a[mij][n - mij + 1] > val then
            dr ← mij - 1
        Else
            st ← mij + 1
        EndIf
    EndWhile
    Return 0
EndSubalgorithm

```

Precizati care dintre urmatoarele afirmatii sunt adevarate cu privire la subalgoritmul *ce_face*(a, n):

- A. Returneaza 1 daca si numai daca valoarea **val** apare in matrice.

- B. Algoritmul verifica daca valoarea **val** apare pe diagonala secundara din matrice.
- C. Algoritmul nu verifica corect daca valoarea **val** apare pe diagonala secundara intrucat elementele acesteia nu sunt ordonate crescator.
- D. Algoritmul returneaza mereu 0.

8. Se considera subalgoritmul *sortare* care primeste ca si parametri de intrare un sir a cu n elemente numere naturale nenule ($1 \leq n \leq 10^6$, $1 \leq a[i] \leq 10^9$, $1 \leq i \leq n$):

Subalgorithm *sortare*(a, n):

For $i \leftarrow 1, n - 1$ **execute**

For $j \leftarrow i + 1, n$ **execute**

If $\text{prim}(a[i])$ **AND** $\text{prim}(a[j])$ **AND** $a[i] < a[j]$ **then**

$a[i] \leftarrow a[i] + a[j]$

$a[j] \leftarrow a[i] - a[j]$

$a[i] \leftarrow a[i] - a[j]$

EndIf

EndFor

EndFor

EndSubalgorithm

Care dintre urmatoarele afirmatii sunt adevarate considerand ca functia *prim* verifica corect daca un numar este prim?

- A. Subalgoritmul sorteaza elementele sirului in asa fel incat cele prime sa ajunga pe primele pozitii.
- B. Subalgoritmul sorteaza in ordine descrescatoare elementele prime din sir, pastrand elementele care nu sunt prime pe pozitiiile lor initiale.
- C. Subalgoritmul sorteaza corect in ordine descrescatoare elementele sirului.
- D. Subalgoritmul sorteaza elementele prime in ordine crescatoare, pastrand elementele care nu sunt prime pe pozitiiile lor initiale.

9. Care afirmatii sunt adevarate?

- A. $2122021021(3) = 78237(9)$
- B. $24321(5) = 5232(7)$
- C. $2436(9) = 11100010111(2)$
- D. $1010111110101001010(2) = \text{AF54A}(16)$

10. Precizati care dintre urmatoarele perechi reprezinta cel mai mic si cel mai mare numar care poate fi stocat pe un tip de date intreg pentru care au fost alocati 20 de biti:

- A. $(-1.048.576, 1.048.575)$
- B. $(-1.048.575, 1.048.575)$
- C. $(-524.288, 524.287)$
- D. $(-524.287, 524.287)$

11. Se considera subalgoritmul *rec* care primește ca și parametrii de intrare un sir **a** cu **n** elemente numere naturale nenule și valoarea **n** și returnează o valoare număr natural ($1 \leq n \leq 10^3$, $1 \leq a[i] \leq 10^9$, $1 \leq i \leq n$).

```
Subalgorithm rec(a, n):  
    If n = 1 then  
        Return a[n]  
    EndIf  
    val  $\leftarrow$  rec(a, n-1)  
    prod  $\leftarrow$  val * a[n]  
    While val  $\neq$  0 then  
        r  $\leftarrow$  a[n] MOD val  
        a[n]  $\leftarrow$  val  
        val  $\leftarrow$  r  
    EndWhile  
    Return prod DIV a[n]  
EndSubalgorithm
```

Precizați care dintre următoarele afirmații sunt adevărate referitoare la subalgoritmul *rec*(a, n):

- A. Subalgoritmul calculează cel mai mare divizor comun al elementelor din sir.
 - B. Subalgoritmul are o complexitate liniară ($O(n)$).
 - C. Subalgoritmul determină cel mai mic multiplu comun al elementelor din sir.
 - D. Subalgoritmul nu are o complexitate liniară ($O(n)$).
12. Precizați pentru ce valori ale lui A, B și C, expresia următoare este adevărată:
A AND B OR !(B OR C) AND A OR B AND C
- A. $A \leftarrow \text{TRUE}$, $B \leftarrow \text{TRUE}$, $C \leftarrow \text{TRUE}$
 - B. $A \leftarrow \text{FALSE}$, $B \leftarrow \text{FALSE}$, $C \leftarrow \text{TRUE}$
 - C. Orice valori pentru A, B, C
 - D. Niciuna dintre variantele de răspuns de mai sus nu este corectă.

13. Se considera subalgoritmul *verificare* care primește ca și parametrii de intrare 3 numere naturale nenule **a**, **b** și **n** ($1 \leq a, b, n \leq 10^6$):

```
Subalgorithm verificare(a, b, n):  
    If a = b then  
        Return a * a = n  
    EndIf  
    mij  $\leftarrow$  (a + b) DIV 2  
    Return verificare(a, mij, n) OR verificare(mij+1, b, n)  
EndSubalgorithm
```

Știind că subalgoritmul este apelat sub forma *verificare*(1, n, n), care dintre următoarele afirmații sunt adevărate?

- A. Algoritmul verifica daca n este patrat perfect intr-o complexitate logaritmica.
- B. Algoritmul verifica daca n este patrat perfect intr-o complexitate liniara.
- C. Algoritmul cauta binar radacina numarului n .
- D. Algoritmul determina si returneaza radacina numarului n .

14. Se considera subalgoritmul *cifre* care primeste ca si parametru unic de intrare un numar natural nenul n care contine doar cifre nenule ($1 \leq n \leq 10^6$).

Subalgorithm cifre(n):

$\text{cnt1} \leftarrow 0$

$\text{cnt2} \leftarrow 0$

While $n \neq 0$ **execute**

If $n \bmod 2 = 0$ **then**

$\text{cnt2} \leftarrow \text{cnt2} + 1$

Else

$\text{cnt1} \leftarrow \text{cnt1} + 1$

EndIf

$n \leftarrow n \text{ DIV } 10$

EndWhile

If $\text{cnt1} > \text{cnt2}$ **then**

Return $\text{cnt1} - \text{cnt2} < 2$

Else

Return $\text{cnt2} - \text{cnt1} < 2$

EndIf

EndSubalgorithm

Care dintre urmatoarele afirmatii sunt adevarate?

- A. Subalgoritmul verifica daca cifrele numarului n pot fi rearanjate in asa fel incat sa nu avem 2 cifre de aceiasi paritate alaturate.
- B. Subalgoritmul verifica daca diferenta dintre numarul de cifre de pe pozitii pare si impare din numar este mai mica decat 2.
- C. Subalgoritmul verifica daca valoarea absoluta a diferentei dintre numarul de cifre pare si numarul de cifre impare ale numarului n este mai mica decat 2.
- D. Subalgoritmul returneaza diferenta dintre numarul de cifre pare si numarul de cifre impare ale numarului n .

15. Se considera subalgoritmul *suma* care primeste ca si parametrii de intrare un sir a cu n elemente numere naturale nenule si numarul natural n ($1 \leq n \leq 10^2$, $1 \leq a[i] \leq 10^9$, $1 \leq i \leq n$):

Subalgorithm suma(a, n):

If $n \leq 0$ **then**

Return 0

EndIf

$\text{sum1} \leftarrow \text{suma}(a, n-1)$

$\text{sum2} \leftarrow \text{suma}(a, n-2)$

$\text{sum3} \leftarrow \text{suma}(a, n-3) + a[n]$

```
    If sum1 ≥ sum2 AND sum1 ≥ sum3 then
        Return sum1
    Else
        If sum2 ≥ sum1 AND sum2 ≥ sum3 then
            Return sum2
        Else
            Return sum3
        EndIf
    EndIf
EndSubalgorithm
```

Ce se afiseaza pentru apelul suma([1, 3, 2, 5, 1, 7, 3, 4, 1, 2, 11, 5, 3], 13):

- A. 21
- B. 33
- C. 48
- D. 24

16. Scriem un cod backtracking care incearca sa determine o parola despre care stim ca este formata din 4 caractere litere mici ale alfabetului englez si cifre. Stim ca parola contine cel putin o cifra si cel putin o litera. Precizati care este numarul maxim de incercari in care am putea sa ghicim aceasta parola:

- A. 1.212.650
- B. 1.212.640
- C. 1.212.641
- D. 1.212.651

17. Se considera subalgoritmul *back* care primeste ca si parametrii de intrare un sir **sol**, un numar natural **poz** si un numar natural **n** ($1 \leq \text{poz} \leq n$, $1 \leq n \leq 10^2$):

```
Subalgorithm back(sol, poz, n):
    For i ← 1, n execute
        If poz != i then
            sol[poz] ← i
            ok ← 1
            For j ← 1, poz - 1 execute
                If sol[j] = sol[poz] then
                    ok ← 0
                EndIf
            EndFor
            If ok = 1 then
                If poz = n then
                    write 1, ' '
                Else
                    back(sol, poz + 1, n)
                EndIf
            EndIf
        EndIf
    EndFor
EndSubalgorithm
```


Considerand de fiecare data apelul initial sub forma *back(sol, 1, n)*, care dintre urmatoarele afirmatii sunt adevarate?

- A. Pentru $n = 5$, se afiseaza pe ecran 120 de valori.
- B. Pentru $n = 4$, se afiseaza pe ecran 9 valori.
- C. Pentru $n = 6$, se afiseaza pe ecran 265 de valori.
- D. Pentru $n = 6$, se afiseaza pe ecran 720 de valori.

18. Se considera subalgoritmul *f* care primeste ca si parametrii de intrare 3 numere naturale nenule ($1 \leq a, b, c \leq 10^6$).

Subalgoritm *f*(*a*, *b*, *c*):

```
For i ← 1, a execute
  cnt ← 1
  For j ← 1, b DIV i execute
    cnt ← cnt + 1
  EndFor
  While cnt > 1 execute
    cnt ← cnt - 1
  EndWhile
  For j ← 1, c execute
    cnt ← cnt + 1
  EndFor
EndFor
EndSubalgorithm
```

Precizati care este complexitatea timp a acestui subalgoritm.

- A. $O(\log(a) * b + a * c)$
- B. $O(a * \log(b) + a * c)$
- C. $O(\log(a) * (b + c * a / \log(a)))$
- D. $O(a * b + a * c)$

19. Se considera subalgoritmul *genMat* care primeste ca si parametrii o matrice *a* cu *n* linii si *n* coloane si valoare *n* numar natural nenul ($1 \leq n \leq 10^3$, $1 \leq a[i][j] \leq 10^9$, $1 \leq i, j \leq n$).

Subalgorithm *genMat*(*a*, *n*):

```
For i ← 1, n execute
  For j ← 1, n execute
    If |i - j| < |n + 1 - i - j| then
      a[i][j] ← |i - j|
    Else
      a[i][j] ← |n + 1 - i - j|
    EndIf
  EndFor
EndFor
EndSubalgorithm
```

Precizati care dintre urmatoarele afirmatii sunt adevarate considerand apelul initial *genMat*(*a*, 5) considerand ca $|x|$ este modulul lui *x*:

- A. Suma elementelor din matrice este 20.
- B. Suma elementelor din matrice este 30.
- C. Algoritmul genereaza o matrice patratica in care fiecare element este o valoare pozitiva.
- D. Algoritmul genereaza o matrice patratica in care fiecare element este egal cu distanta fata de cea mai apropiata diagonala.

20. Se considera subalgoritmul *construire* care primeste ca parametru unic de intrare un numar natural nenul n ($1 \leq n \leq 10^9$)

Subalgorithm *construire*(n):

```
nr ← 1
d ← 2
While  $n > 1$  execute
    While  $n \bmod d = 0$  execute
        If  $nr \bmod d \neq 0$  then
             $nr \leftarrow nr * d$ 
        EndIf
         $n \leftarrow n \text{ DIV } d$ 
    EndWhile
     $d \leftarrow d + 1$ 
```

EndWhile

Return nr

EndSubalgorithm

Precizati care dintre urmatoarele afirmatii sunt adevarate referitoare la subalgoritmul *construire*(n):

- A. Subalgoritmul returneaza valoarea lui n .
- B. Subalgoritmul are o complexitate liniara ($O(n)$).
- C. Subalgoritmul determina cel mai mic numar care are aceiasi factori primi ca n .
- D. Se returneaza valoarea lui n daca si numai daca acesta este numar prim.

21. Se considera subalgoritmul *suma* care primeste ca si parametru unic de intrare un numar natural nenul n ($1 \leq n \leq 10^9$).

Subalgorithm *suma*(n):

If $n = 0$ **then**

Return 1

EndIf

Return $n * suma(n - 1) + n$

EndSubalgorithm

Precizati care afirmatii sunt adevarate:

- A. $suma(10) = 3.628.855$
- B. $suma(n) = n! + n$
- C. $suma(n) = n! + n * (n + 1) / 2$
- D. $suma(4) = 88$

22. Se considera subalgoritmul *calcul* care primește ca si parametri de intrare 2 numere naturale nenule n si m ($1 \leq n, m \leq 10^3$).

```
Subalgorithm calcul(n, m):  
    If  $n = 1$  OR  $m = 1$  then  
        Return 2  
    EndIf  
    Return calcul( $n-1, m$ ) * 2 + calcul( $n, m-1$ )  
EndSubalgorithm
```

Precizati care dintre urmatoarele afirmatii sunt adevarate:

- A. calcul(3, 3) = 34
- B. calcul(2, 5) = 18
- C. calcul(4, 5) = 416
- D. calcul(1, 1) = 1

23. Subalgoritm *calcul* primește 2 numere naturale a si b ca parametri de intrare ($1 \leq a, b \leq 10^3$):

```
Subalgorithm calcul(a, b):  
    If  $b = -1$  then  
        Return 1  
    EndIf  
    Return  $(a - b) * \text{calcul}(a, b-1)$   
EndSubalgorithm
```

Ce se calculeaza?

- A. A_a^b
- B. A_a^{b-1}
- C. A_{a-1}^b
- D. calcul(5, 3) = 120

24. Se considera subalgoritmul *graf* care primește ca si parametri de intrare o matrice a cu n linii si n coloane reprezentand matricea de adiacenta a unui graf neorientat cu n noduri, numarul natural nenul n si un **nod** din acest graf ($1 \leq n \leq 10^3$, $0 \leq a[i][j] \leq 1$, $1 \leq i, j \leq n$, $1 \leq \text{nod} \leq n$):

```
Subalgorithm graf(a, n, nod):  
     $P \leftarrow [0] * (n + 1)$   
     $Q \leftarrow [0] * (n + 1)$   
     $st \leftarrow 1$   
     $dr \leftarrow 1$   
     $Q[dr] \leftarrow \text{nod}$   
     $P[nod] \leftarrow 1$   
    While  $st \leq dr$  execute  
         $nd \leftarrow Q[st]$   
         $st \leftarrow st + 1$   
        For  $i \leftarrow 1, n$  execute  
            If  $P[i] = 0$  AND  $a[nd][i] = 1$  then
```

```
        P[i] ← P[nd] + 1
        dr ← dr + 1
        Q[dr] ← i
    EndIf
EndFor
EndWhile
Return P[n] - 1
EndSubalgorithm
```

Precizati care dintre urmatoarele afirmatii sunt adevarate referitoare la subalgoritmul *graf*:

- A. Subalgoritmul determine cate componente conexe are graful nostru.
- B. Subalgoritmul verifica daca graful este conex.
- C. Subalgoritmul returneaza valoarea -1 daca nu exista niciun lant de la nodul **nod** catre nodul **n**.
- D. Subalgoritmul returneaza lungimea celui mai scurt drum de la nodul **nod** catre nodul **n**.

1	B, C	13	B
2	B	14	A, C
3	C, D	15	A
4	D	16	B
5	A	17	B, C
6	C	18	A, C
7	B	19	A, C, D
8	B	20	B, C
9	A, B, C	21	D
10	C	22	A, B
11	C, D	23	D
12	A	24	C, D