

ACADEMIA ZECELAINFO

Simulare Examen Admitere UBB 2023 - 29 aprilie 2023

Proba scrisa la Informatica

NOTA IMPORTANTA:

In lipsa altor precizari:

- presupuneti ca toate operatiile aritmetice se efectueaza pe tipuri de date nelimitate (nu exista *overflow* / *underflow*)
- numerotarea indicilor tuturor sirurilor / vectorilor incepe de la 1
- toate restrictiile se refera la valorile parametrilor actuali la momentul apelului initial.

1. Se considera algoritmul $exp(n)$, unde n este un numar natural nenul ($1 \leq n \leq 10^9$).

```
1: Algorithm  $exp(n)$ :  
2:   If  $n \leq 0$  then  
3:     return 0  
4:   EndIf  
5:   return  $n * (n - 1) \text{ DIV } 2 + exp(n - 2)$   
6: EndAlgorithm
```

Precizati care dintre urmatoarele afirmatii sunt adevarate.

- A. In urma apelului $exp(5)$ algoritmul returneaza **13**.
- B. Apelul $exp(n)$ calculeaza suma $n*(n-1)/2 + (n-2)*(n-1)/2 + \dots + 0$.
- C. Apelul $exp(n)$ returneaza $n*(n-1)/2$ pentru orice $n < 4$, natural nenul.
- D. Algoritmul ar fi functionat la fel daca linia 2 ar fi inlocuita cu **If** $n = 0$ **then**, cu n numar natural nenul.

2. Se considera urmatorul algoritmul $alg(s, a, b)$ unde s este un sir de numere naturale nenule ($s[1], s[2], \dots, s[n]$, $1 \leq s[i] \leq 10^9$, $i = 1, 2, \dots, n$), iar a si b sunt 2 numere naturale cu ($1 \leq a \leq b \leq$ lungimea sirului s).

```
1: Algorithm  $alg(s, a, b)$ :  
2:   If  $a = b$  then  
3:      $x \leftarrow 1$ 
```

```

4:      y ← s[a]
5:      g ← 0
6:      While x ≤ y execute
7:          z ← (2 * x + y - x) DIV 2
8:          If z * z = s[b] then
9:              g ← 1
10:             x ← y + 1
11:          Elseif z * z > s[b] then
12:              y ← z - 1
13:          Else
14:              x ← z + 1
15:          Endif
16:      EndWhile
17:      return g
18:  Else
19:      return alg(s, a, a + (b - a) / 2) + alg(s, a + (b - a) / 2 + 1, b)
20:  Endif
21: EndAlgorithm

```

Pentru care dintre urmatoarele apeluri se returneaza valoarea **5**?

- A. $alg([3, 2, 4, 121, 32, 17, 25, 1, 34, 52, 36], 1, 10)$
- B. $alg([4, 5, 12, 400, 625, 11, 3, 7, 23], 1, 9)$
- C. $alg([1, 2, 3, 4, 5, 6, 7, 8, 9, 10], 1, 10)$
- D. $alg([16, 23, 41, 9, 4, 81, 1, 15], 1, 8)$

3. Se considera algoritmul $f(a, b)$, unde a si b sunt numere naturale ($a > 1, b > 1$).

```

1: Algorithm f(a, b):
2:   r ← 1
3:   While b > 0 execute
4:       If b MOD 2 = 1 then
5:           r ← r * a
6:       Endif
7:       a ← a * a
8:       b ← b / 2
9:   EndWhile
10:  return r
11: EndAlgorithm

```

Alegeti afirmatiile corecte despre algoritmul $f(a, b)$:

- A. Algoritmul $f(a, b)$ va returna o putere a lui 2 atunci cand b este o putere a lui 2.
- B. Algoritmul are o complexitate logaritmica.
- C. Pentru $a = 100$, $b = 2^{15}$, algoritmul va returna un numar cu mai mult de 2^{16} cifre.
- D. Pentru $a = 100.000.000$ si $b = 2^{60}$, algoritmul va returna un numar cu $2^{63} + 1$ cifre.

4. Precizati care dintre urmatoarele formule calculeaza corect cate numere pare sunt in intervalul $[a, b]$ (a, b - numere naturale nenule).

- A. $(b - a + 2) / 2$, daca a si b sunt pare.
- B. $(b - a + 1) / 2$, daca a si b sunt impare.
- C. $(b - a) / 2$, daca a si b sunt impare.
- D. $[b - a + (a \bmod 2 \neq b \bmod 2) + 2 * (a \bmod 2 = b \bmod 2 \text{ AND } a \bmod 2 = 0)] / 2$, oricare ar fi a si b numere naturale.

5. Se considera urmatorul program recursiv de afisare $afis(i, n)$ unde i si n reprezinta 2 numere naturale cu $i \leq n$.

```
1: Algorithm  $afis(i, n)$ :
2:   If  $i \leq n$  then
3:     For  $nr \leftarrow 2^{i-1} + 1, \quad 2^{i-1} + 2^{i-2}$  execute
4:       Write  $nr, ' '$ 
5:     EndFor
6:      $afis(i+1, n)$ 
7:     For  $nr \leftarrow 2^{i-1} + 2^{i-2} + 1, \quad 2^i$  execute
8:       Write  $nr, ' '$ 
9:     EndFor
10:  EndIf
11: EndAlgorithm
```

Fie urmatoarea secventa de cod care apeleaza functia $afis$:

```
1: Write 1, ' '
2:  $afis(2, n)$ 
3: Write 2, ' '
```

Precizati care dintre urmatoarele afirmatii sunt adevarate in urma executarii celor 3 linii de cod de deasupra.

- A. Pentru $n = 3$, se afiseaza 1 3 4 5 6 7 8 2
- B. Pentru $n = 3$, se afiseaza 1 3 5 6 7 8 4 2
- C. Pentru orice n , la mijlocul şirului format din primele $2*i$ numere naturale se inserează şirul ordonat format din următoarele $2*i$ numere naturale unde i ia valori de la 1 la n .
- D. Niciuna dintre afirmatii NU este adevarata.

6. Precizati care dintre urmatoarele afirmatii referitoare la egalitatea unor valori in baze de numeratie diferite sunt adevarate:

- A. $10010101111010100101000101111_{(2)} = 12BD4A2F_{(16)}$
- B. $7383256_{(9)} = 21102210021220_{(3)}$
- C. $23020021011021002_{(3)} = 296234232_{(9)}$
- D. $10110110_{(2)} = 2321_{(4)}$

7. Se considera algoritmul *ceFace*(sir, a, b, v), unde **sir** este un vector format din n ($1 \leq n \leq 1000$) numere naturale nenule distincte ordonate crescator (**sir**[1], **sir**[2], ..., **sir**[n]), **a**, **b** si **v** sunt numere naturale ($1 \leq a, b \leq n$).

```
1: Algorithm ceFace(sir, a, b, v):
2:   If a > b then
3:     return a
4:   EndIf
5:    $c \leftarrow (a + b) / 2$ 
6:   If sir[c]  $\geq v$  then
7:     return ceFace(sir, a, c-1, v)
8:   Else
9:     return ceFace(sir, c+1, b, v)
10:  EndIf
11: EndAlgorithm
```

Care dintre urmatoarele afirmatii sunt adevarate, considerand ca apelul initial este *ceFace*(sir, 1, n, v)?

- A. Algoritmul returneaza cel mai mare element, mai mic sau egal cu valoarea **v** care apare in sirul sir.
- B. Algoritmul returneaza cel mai mic element, mai mare sau egal cu valoarea **v** care apare in sirul sir.

- C. In cazul in care v este mai mare decat ultimul element din sir, atunci se returneaza valoarea $n+1$.
- D. In cazul in care v nu apare in sir, algoritmul va intra intr-o bucla infinita.

8. Se considera algoritmul $f(a, b)$, unde a si b sunt numere naturale nenule ($1 \leq a, b \leq 10^9$).

```

1: Algorithm  $f(a, b)$ :
2:   If  $b = 0$  then
3:     return  $a$ 
4:   EndIf
5:   return  $f(b, a \bmod b)$ 
6: EndAlgorithm

```

Precizati care dintre urmatoarele afirmatii sunt false.

- A. In urma apelului $f(674, 97)$ algoritmul returneaza 1.
- B. Exista cazuri cand algoritmul nu isi termina executia din cauza conditiei de pe linia 2.
- C. Complexitatea acestui algoritm este $O(\log_2(a + b))$.
- D. Algoritmul calculeaza numarul de divizori ai numarului a .

9. Se considera algoritmul $transformare(n)$, unde n este un numar natural nenul ($1 \leq n \leq 10^9$).

```

1: Algorithm  $transformare(n)$ :
2:   If  $n = 0$  then
3:     return 0
4:   EndIf
5:    $c \leftarrow n \bmod 10$ 
6:    $ok \leftarrow 1$ 
7:   If  $c \leq 1$  OR  $c = 9$  then
8:      $ok \leftarrow 0$ 
9:   EndIf
10:  If  $c \bmod 2 = 0$  AND  $c \neq 2$  then
11:     $ok \leftarrow 0$ 
12:  EndIf
13:  If  $ok = 1$  then
14:    return  $transformare(n \text{ DIV } 10) * 100 + c * 10 + c$ 

```

```

15:  Else
16:      return transformare(n DIV 10) * 10 + c
17:  EndIf
18:EndAlgorithm

```

Precizati care dintre urmatoarele afirmatii sunt corecte.

- A. Apelul *transformare*(357) va returna 335577
- B. Apelul *transformare*(2357) va returna 77553322
- C. Algoritmul *transformare*(*n*) dubleaza toate cifrele impare din numarul *n* cu exceptia cifrei 1.
- D. Algoritmul *transformare*(*n*) dubleaza toate cifrele prime din numarul *n* si returneaza numarul in ordine inversa.

10. Se considera subalgoritmul *ceFace*(*a*, *n*, *i*, *j*) unde *a* este un sir cu *n* elemente numere naturale nenule (*a*[1], *a*[2], ..., *a*[*n*]), iar *n* este lungimea acestui sir. Variabilele *i* si *j* sunt 2 numere naturale nenule, initializate la apelul initial cu valoarea 1.

```

1:Algorithm ceFace(a, n, i, j):
2:    If i < n then
3:        If j ≥ 2 then
4:            If a[j] < a[j-1] then
5:                a[j] ↔ a[j-1]
6:            EndIf
7:            ceFace(a, n, i, j-1)
8:        Else
9:            ceFace(a, n, i+1, i+1)
10:    EndIf
11: EndIf
12:EndAlgorithm

```

Precizati care dintre urmatoarele afirmatii referitoare la subalgoritmul *ceFace*(*a*, *n*, *i*, *j*) sunt adevarate.

- A. Algoritmul amesteca elementele sirului *a* fara nicio logica.
- B. Algoritmul efectueaza o sortare crescatoare utilizand algoritmul Sortarii prin Selectie.
- C. Algoritmul efectueaza o sortare descrescatoare utilizand algoritmul Sortarii prin Selectie.

D. Algoritmul sorteaza crescator elementele sirului **a**, utilizand algoritmul de Sortare prin Insertie.

11. Se considera algoritmul $nr(n)$ care primeste ca si parametru un numar natural nenul n ($1 \leq n \leq 1000$).

```
1: Algorithm  $nr(n)$ :  
2:    $d \leftarrow 2$   
3:    $rez \leftarrow 1$   
4:   While  $n > 1$  execute  
5:      $p \leftarrow 0$   
6:     While  $n \bmod d = 0$  execute  
7:        $p \leftarrow p + 1$   
8:        $n \leftarrow n \text{ DIV } d$   
9:     EndWhile  
10:    If  $d = 2$  then  
11:       $rez \leftarrow rez * p$   
12:    Else  
13:       $rez \leftarrow rez * (p + 1)$   
14:    EndIf  
15:     $d \leftarrow d + 1$   
16:    If  $d * d > n$  then  
17:       $d \leftarrow n$   
18:    EndIf  
19:  EndWhile  
20:  return  $rez$   
21: EndAlgorithm
```

Precizati care dintre urmatoarele afirmatii sunt adevarate.

- A. Algoritmul calculeaza numarul de divizori ai numarului **n**.
- B. Algoritmul descompune in factori primi numarul **n**.
- C. Algoritmul calculeaza numarul de divizori impari ai numarului **n**.
- D. Algoritmul calculeaza numarul de divizori pari ai numarului **n**.

12. Precizati care dintre urmatoarele expresii au valoarea adevarat **daca si numai daca** numarul natural nenul **n** este multiplu de 6, dar nu este multiplu de 12.

- A. $(n \bmod 2 = 0) \text{ AND } (n \bmod 3 = 0) \text{ AND } (n \bmod 4 \neq 0)$
- B. $(n \bmod 6 = 0) \text{ AND } (n \bmod 12 \neq 0)$

C. $(n + 6) \bmod 12 = 0$

D. $(n \bmod 4 \neq 0) \text{ AND } (n \bmod 2 = 0) \text{ AND } (n \bmod 3 \neq 0)$

13. Care este complexitatea urmatoarei secvente de cod?

```

1: Algorithm fct(n):
2:    $i \leftarrow 1$ 
3:   While  $i < n$  execute
4:      $j \leftarrow \sqrt{i}$ 
5:     While  $j > 0$  execute
6:        $j \leftarrow j \text{ DIV } 2$ 
7:     EndWhile
8:      $i \leftarrow i + 1$ 
9:   EndWhile
10: EndAlgorithm

```

A. Complexitatea este $O(n * \log_2 n)$

B. Complexitatea este $O(\lg(\sqrt{n!}))$

C. Complexitatea este $O(n * \lg(\sqrt{n}))$

D. Complexitatea este $O(\log_2 \sqrt{n!})$

14. Fie o stivă inițial vidă. Cu ajutorul subprogramelor *push*(x) și *pop*() este adăugat elementul **x**, respectiv șters un element din stivă. Suma elementelor din stivă după executarea operațiilor următoare este:

push(5), *pop*(), *push*(3), *push*(7), *push*(23), *pop*(), *push*(11), *pop*(), *pop*(), *push*(12), *push*(15)

A. 30

B. 37

C. 27

D. Niciun raspuns nu este corect.

15. Se considera subprogramul *nr_mod*(n, a, divi, nrdiv), unde **n** este un numar natural nenul, iar **a** este un parametru care la apelul initial al functiei este

initializat cu 1. De asemenea, sirul **divi** este un sir care contine toti divizorii numarului **n** in ordine crescatoare si are lungimea **nrdiv**.

```
1: Algorithm nr_mod(n, a, divi, nrdiv):
2:   If n = 0 then
3:     return 1
4:   EndIf
5:   s ← 0
6:   i ← a
7:   While i ≤ nrdiv AND divi[i] ≤ n execute
8:     s ← s + nr_mod(n - divi[i], i, divi, nrdiv)
9:     i ← i + 1
10:  EndWhile
11:  return s
12: EndAlgorithm
```

Precizati care dintre urmatoarele afirmatii sunt corecte.

- A. Algoritmul calculeaza numarul de moduri in care poate fi scris **n** ca suma de divizori distincti ai sai.
- B. Algoritmul calculeaza numarul de moduri in care poate fi scris **n** ca suma de divizori, nu neaparat distincti, ai sai.
- C. Algoritmul calculeaza suma tuturor divizorilor lui **n**.
- D. Pentru **n** = 9, se returneaza valoarea 5.

16. Precizati cate numere de 5 cifre au proprietatea ca nu contin cifre prime pe pozitii consecutive. Ex. 12345 nu este valid intrucat contine 2,3 pe pozitii consecutive, dar 26983 este valid.

- A. 2364
- B. 5184
- C. 50256
- D. Niciun raspuns nu este corect.

17. Se considera urmatorul sir $s = 1, 1, 4, 1, 4, 9, 1, 4, 9, 16, \dots$. Acest sir se genereaza prin alipirea la fiecare pas a unui sir format din primele **i** patrate perfecte cu **i** luand pe rand valorile 1, 2, 3, Precizati care dintre urmatoarele afirmatii sunt adevarate:

- A. Al 234-lea termen al sirului este 16.
- B. Al 23-lea termen al sirului este 4.

- C. Al 500-lea termen al sirului este 16.
- D. Al 1043-lea termen al sirului este 49.

18. Se da subalgoritmul $verif(n)$ care primeste ca si parametru un numar natural n .

```

1: Algorithm  $verif(n)$ :
2:    $nrd \leftarrow 1$ 
3:    $d \leftarrow 2$ 
4:   While  $n > 1$  execute
5:     If  $n \bmod d = 0$  then
6:       return 0
7:     EndIf
8:      $d \leftarrow d + 1$ 
9:     If  $d * d > n$  then
10:      return 1
11:    EndIf
12:  EndWhile
13:  return 1
14: EndAlgorithm

```

Precizati care dintre urmatoarele afirmatii sunt **false**.

- A. Algoritmul verifica corect daca numarul n este prim.
- B. Algoritmul verifica corect daca numarul n este prim, doar daca valoarea lui este mai mare sau egala cu 2.
- C. Algoritmul nu verifica corect daca numarul n este prim.
- D. Daca $n > 2$, algoritmul verifica corect daca numarul este prim.

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

```

1: Algorithm  $f(n)$ :
2:   If  $n = 0$  then
3:     return 10
4:   EndIf
5:    $c \leftarrow n \bmod 10$ 
6:    $m \leftarrow f(n \text{ DIV } 10)$ 
7:   If  $c < m$  then

```

```

8:         return c
9:     Else
10:         return m
11:     EndIf
12:EndAlgorithm

```

Precizati care dintre urmatoarele afirmatii sunt **adevarate** referitoare la acest subalgoritm.

- A. $f(4682363846)$ returneaza 2.
- B. $f(237476327)$ returneaza 7.
- C. Functia determina cea mai mica cifra a numarului n .
- D. Functia determina cifra maxima a numarului n .

20. Se considera algoritmul *matrice*(a, n) care primeste ca si parametrii o matrice patratica cu n linii si n coloane cu n numar natural nenul ($1 \leq n \leq 100$, $a[1][1], a[1][2], \dots, a[1][n], a[2][1], \dots, a[2][n], \dots, a[n][n]$, $1 \leq a[i][j] \leq 10^9$, oricare ar fi $1 \leq i, j \leq n$) si $\max(a, b)$ care primeste ca si parametrii 2 numere naturale si returneaza valoarea maxima.

```

1:Algorithm matrice(a, n):
2:    s ← 0
3:    For i ← 2, n - 1 execute
4:        For j ← max(n - i + 1, i) + 1, n execute
5:            s ← s + a[i][j]
6:        EndFor
7:    EndFor
8:    Return s
9:EndAlgorithm

```

Care dintre urmatoarele afirmatii sunt **adevarate** in legatura cu subalgoritmul de mai sus?

- A. Calculeaza suma elementelor situate deasupra de diagonala principala si deasupra de diagonala secundara.
- B. Calculeaza suma elementelor situate deasupra de diagonala principala si sub diagonala secundara.
- C. Calculeaza suma elementelor situate sub diagonala principala si deasupra de diagonala secundara.

D. Calculeaza suma elementelor situate sub diagonala principala si sub diagonala secundara.

21. Folosind metoda backtracking se generează toate șirurile formate din patru caractere distincte din mulțimea $\{#, *, \&, @, \%\}$. Primele trei soluții sunt: $\#*\&@$, $\#*\&\%$, $\#* @\&$. Indicați care este soluția generată înainte de $\&*#@$.

- A. $\&\#@ \%$
- B. $*\&@ \%$
- C. $\&@ \# \%$
- D. $\&\# \% @$

22. Se considera algoritmul $magic(n, a)$, unde a este un vector cu n numere naturale($a[1], a[2], \dots, a[n], 1 \leq n \leq 10000$).

```
1:Algorithm magic(n, a):
2:   If  $n < 2$  then
3:     return True
4:   EndIf
5:   For  $i \leftarrow 2, n$  execute
6:     If  $a[i-1] \neq a[i]$  then
7:       return False
8:     EndIf
9:   EndFor
10:  return True
11:EndAlgorithm
```

Precizati care dintre urmatoarele afirmatii sunt adevarate:

- A. Pentru $magic(5, [4, 4, 4, 4, 4])$ algoritmul returneaza False.
- B. Pentru $magic(7, [3, 2, 4, 5, 2, 3, 1])$ algoritmul returneaza True.
- C. Oricare ar fi un sir cu n elemente identice, algoritmul returneaza valoarea True.
- D. In cazul in care algoritmul returneaza False, vom sti ca in sir nu exista 2 valori egale pe pozitii consecutive.

23. Se considera subalgoritmii $f1(n)$ care primeste ca si parametru un numar natural nenul n si $f2(a, n)$ care primeste ca si parametrii un sir a cu n elemente numere naturale nenule ($1 \leq n \leq 10^6, a[1], a[2], \dots, a[n]$).

```
1:Algorithm f1(n):
```

```

2:   If  $n = 0$  OR  $n = 1$  then
3:       return 0
4:   EndIf
5:   If  $n \bmod 2 = 0$  AND  $n \neq 2$  then
6:       return 0
7:   EndIf
8:   For  $i \leftarrow 3, \sqrt{n}, 2$  execute
9:       If  $n \bmod i = 0$  then
10:          return 0
11:       EndIf
12:   EndFor
13:   return 1
14:EndAlgorithm

```

```

1:Algorithm  $f2(a, n)$ :
2:    $st \leftarrow 1$ 
3:    $dr \leftarrow n$ 
4:    $poz \leftarrow 1$ 
5:   For  $i \leftarrow st, dr$  execute
6:       If  $f1(a[i]) = 1$  then
7:            $a[i] \leftrightarrow a[poz]$ 
8:            $poz \leftarrow poz + 1$ 
9:       EndIf
10:  EndFor
11:EndAlgorithm

```

Precizati care dintre urmatoarele afirmatii sunt adevarate referitor la subalgoritmul $f2$:

- A. Subalgoritmul suprascrive primele elemente ale vectorului a , inlocuindu-le cu valorile prime din vector.
- B. Subalgoritmul sterge toate elementele prime din vector.
- C. Subalgoritmul rearanjeaza elementele sirului in asa fel incat cele prime sa ocupe primele pozitii ale sirului, iar cele care nu sunt prime sa fie in continuarea acestora.
- D. Ordinea elementelor prime ramane neschimbata.

24. In legatura cu subalgoritmul $f1$ de la programul anterior, care dintre urmatoarele afirmatii sunt corecte?

- A. Algoritmul are o complexitate logaritmica.

- B. Algoritmul verifica daca numarul n este prim oricare ar fi n un numar natural.
- C. Algoritmul functioneaza chiar daca numarul n este negativ.
- D. Nu exista niciun alt algoritm de verificare a primalitatii care sa aiba o complexitate mai buna.

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

Felicitari pentru rezolvarea completa a acestui subiect!

©ACADEMIA ZECELAINFO, 2023