Universitatea din Bucureşti
Facultatea de Matematică-Informatică
Calculabilitate şi Complexitate
Examen, 7 Februarie, Nivelul I, Subiecte: A

Nume: ______________________

Puncte:26. Timp: 52 min
Data: 10-05-2023

Instrucțiuni I. Incercuiți rãspunsul/rãspunsurile corecte. Toate intrebãrile conteazã in mod egal. Puteți folosi marginile hârtiei drept ciornã, **dar nicun alt material**. O intrebare poate avea mai multe rãspunsuri corecte. Toate conteazã in mod egal. **Pe de altã parte, dacã alegeți un rãspuns greșit, punctajul vostru la intrebare este zero.**

1. Funcţia lui Ackermann este
   (a) primitiv recursivã.
   (b) recursivã.
   (c) parţial recursivã.
   (d) niciunul dintre celelalte rãspunsuri nu este corect.
2. Care operaţii sunt folosite in definiţia unei funcţii primitiv recursive ?
   (a) compunere
   (b) recursie primitivã
   (c) minimizare
   (d) exponenţiere
3. O funcţie parţial recursivã este recursivã doar dacã ...
   (a) este obţinutã prin minimizare.
   (b) este definitã in orice punct x.
   (c) poate fi calculatã de o maşinã Turing.
   (d) creşte mai rapid decât orice funcţie primitiv recursivã.
4. Cum putem crea o funcţie care este primitiv recursivã ?
   (a) enumerãm toate funcţiile primitiv recursive. Creãm o funcţie care pe inputul i returneazã valoarea $f_i(i) + 1$.
   (b) folosim operacţia de minimizare, dacã rezultatul nu este definitã pentru toate inputurile i.
   (c) prin compunere din douã funcţii primitiv recursive.
   (d) Cu un automat finit.

5. Dacã A este o mulţime recursiv enumerabilã iar B este problema de a testa dacã un numãr binar este prim, atunci
   (a) $A \leq_m B$.
   (b) $B \leq_m A$.
   (c) Ambele reduceri sunt adevãrate.
   (d) Nicio reducere nu e adevãratã.
6. Dacã A, B sunt probleme de decizie iar $A \oplus B = \{x0|x \in A\} \cup \{y1|y \in B\}$ atunci
   (a) $A \leq_m A \oplus B$.
   (b) $A \oplus B \leq_m A$.
   (c) Ambele reduceri sunt adevãrate.
   (d) Nicio reducere nu e adevãratã
7. Care din urmãtoarele de calcul recunoaşte cea mai micã clasã de limbaje ?
   (a) automat finit
   (b) maşinã Turing cu o bandã.
   (c) maşinã Turing cu douã benzi.
   (d) Toate modelele menţionate recunosc aceeaşi clasã de limbaje.
8. O maşinã Turing universalã ...
   (a) poate simula orice maşinã Turing.
   (b) acceptã un limbaj recursiv enumerabil.
   (c) are o bandã specialã, numitã *bandã oracol*.
   (d) se opreşte intotdeauna intr-un numãr finit de paşi.
9. Care din problemele urmãtoare *nu* sunt recursive ?
   (a) problema opririi K.
   (b) problema de a decide dacã un numãr este prim.
   (c) problema satisfiabilitãţii.
   (d) Niciuna din problemele listate.
10. Care din clasele urmãtoare de probleme de decizie sunt inchise la operaţia de complementare (dacã $A \in \mathcal{C}$ atunci $\overline{A} \in \mathcal{C}$) ?
   (a) clasa problemelor recursive
   (b) clasa problemelor primitiv recursive.
   (c) clasa problemelor parţial recursive.
   (d) P clasa problemelor care au algoritmi poliniomiali.

11. Care din problemele următoare **nu** sunt recursive ?

   (a) $K_1 = \{< x, y >: M_x(y)$ se opreşte intr-un pas$\}$

   (b) $K = \{< x, y >: M_x(y)$ se opreşte$\}$.

   (c) $\overline{K} = \{< x, y >: M_x(y)$ nu se opreşte$\}$

   (d) toate problemele sunt recursive.

12. Care din următoarele probleme sunt nedecidabile ?

   (a) Fiind date o mulţime de tipuri de pavaje, problema dacă putem pava planul cu pavajele Wang date.

   (b) Fiind date o mulţime de tipuri de pavaje, problema dacă putem pava un pătrat 1000x1000 cu pavajele Wang date

   (c) Fiind dat un polinom cu coeficienţi intregi $p(x_1, x_2, \ldots, x_n)$ există soluţii intregi pentru ecuaţia $p(x_1, x_2, \ldots, x_n) = 0$?

   (d) Fiind dat un polinom cu coeficienţi intregi $p(x_1, x_2, \ldots, x_n)$ există soluţii pentru ecuaţia $p(x_1, x_2, \ldots, x_n) = 0$ cu $|x_1|, \ldots, |x_n| \leq 1000$?

13. Un exemplu de funcţie care *nu* este parţial recursivă este:

   (a) Funcţia lui Ackermann

   (b) Funcţia "Busy Beaver"

   (c) Functia $f(x)$ care numără paşii făcuţi de o maşină Turing universală M_U pe intrarea x (nedefinită dacă M_U nu se opreşte)

   (d) Toate funcţiile de mai sus sunt parţial recursive.

14. Fiind dată o maşină Turing deterministă, care din afirmaţiile următoare sunt adevărate?

   (a) Dacă pe o intrare x maşina rulează in $f(|x|)$ paşi, atunci pe orice intrare y spaţiul folosit de maşină este $O(f(|y|))$.

   (b) Dacă pe o intrare x maşina rulează in spaţiu $f(|x|)$, atunci pe orice intrare y maşina rulează in $O(f(|y|))$ paşi.

   (c) Dacă pe o intrare x maşina nu se opreşte atunci spaţiul folosit de $M(x)$ este infinit.

   (d) niciuna din afirmaţii nu este adevărată.

15. Fiind dată formula următoare: $x \vee y \vee t, \overline{y} \vee \bar{t}, \overline{z} \vee \bar{t}$, care literali sunt puri ?

   (a) x

   (b) y

   (c) z

   (d) t

16. Care din urmãtoarele afirmaţii sunt adevãrate ?

    (a) Orice algoritm polinomial nedeterminist poate fi simulat de un algoritm determinist cu complexitate polinomialã.

    (b) Orice algoritm polinomial nedeterminist poate fi simulat de un algoritm determinist cu complexitate $O(2^{n^{O(1)}})$.

    (c) Dându-se o formulã booleanã in care toate variabilele sunt cuantificate, existã un algoritm polinomial pentru a decide dacã fomulele sunt adevãrate sau false.

    (d) Pentru o parte a rãspunsurilor de mai sus nu se cunoaşte statutul lor de adevãr.

17. Care din urmãtoarele probleme **nu** sunt cunoscute ca fiind NP-complete ?

    (a) 2-SAT

    (b) 3-SAT

    (c) Problema de a decide dacã un numãr este prim sau nu.

    (d) problema Vertex Cover.

18. Care din urmãtoarele afirmaţii sunt adevãrate ?

    (a) Dacã $A \leq_m^P B$ şi $B \in P$ atunci $A \in P$.

    (b) Dacã $A \leq_m^P B$ şi $B \in NP$ atunci $A \in NP$.

    (c) Dacã $A \leq_m^P B$ şi A este NP-completã atunci B este NP-completã.

    (d) Dacã $A \leq_m^P B$ şi A este NP-hard atunci B este NP-hard.

19. Care din urmãtoarele afirmaţii este adevãratã ?

    (a) O formulã logicã CNF cu cel mult doi literali in fiecare clauzã are demonstraţii prin rezoluţie de lungime polinomialã in n.

    (b) O formulã logicã CNF cu cel mult trei literali in fiecare clauzã are demonstraţii prin rezoluţie de lungime polinomialã in n.

    (c) Existã un algoritm de tip Davis-Putnam care pe instanţe nesatisfiabile pentru 2-SAT ruleazã in timp polinomial in n.

    (d) Existã un algoritm de tip Davis-Putnam care pe instanţe nesatisfiabile pentru 3-SAT ruleazã in timp polinomial in n.

20. Care din urmãtoarele probleme au algoritmi polinomiali ?

    (a) Fiind datã o mulţime de numere, sunt ele in ordine sortatã ?

    (b) Fiind dat un graf orientat G şi douã vârfuri s, t, putem ajunge de la s la t?

    (c) Putem colora un graf cu 3 culori astfel incât orice douã vârfuri adiacente sã aibã culori diferite ?

    (d) Fiind datã o formulã propoziţionalã in forma normalã conjunctivã in care in fiecare clauzã apare cel mult un literal negativ, este formula satisfiabilã ?

21. Dacã $P \neq NP$ atunci ...

(a) Putem colora un graf cu numãrul minim de culori in timp polinomial.

(b) putem rezolva orice problem a de decizie cu un algoritm polinomial.

(c) putem gãsi o soluţie pentru problema Vertex Cover cu un algoritm cu complexitate polinomialã.

(d) Orice problemã rezolvabilã in timp polinomial folosind 2-SAT ca subrutinã are un algoritm polinomial.

22. Care din urmãtoarele clase de complexitate este cunoscutã ca inchisã la complementare ?

(a) co-NP.

(b) toate clasele Σ_k^P din ierarhia polinomialã.

(c) PSPACE.

(d) Niciuna din clase.

23. Teorema lui Matiyasevich ...

(a) Aratã faptul cã SAT este NP-completã.

(b) Simuleazã funcţionarea unei Maşini Turing nedeterministe polinomiale cu o formulã logicã.

(c) Dã un algoritm polinomial pentru SAT.

(d) Niciuna din celelalte opţiuni.

24. Dacã $P \neq NP$ atunci ...

(a) $NP = co - NP$.

(b) Putem gãsi in timp polinomial un martor pentru o instanţã pozitivã a unei probleme din NP.

(c) Orice problemã NP-hard este in P

(d) Nu putem gãsi o clicã maximã intr-un graf in timp polinomial.

25. Un algoritm nedeterminist care foloseşte spaţiu $s(n)$ poate fi simulat de ...

(a) un algoritm determinist care foloseşte spaţiu $O(s^2(n))$.

(b) Un algoritm determinist polinomial.

(c) Un algoritm pentru 3-SAT.

(d) Niciun rãspuns nu e corect.

26. Care probleme NP-complete pot fi folosite in practicã pentru rezolvarea altor probleme NP-complete ?

(a) SAT

(b) ILP, problema programãrii liniare in numere intregi.

(c) problema izomorfismului a douã grafuri

(d) niciuna, nu existã algoritmi practici pentru probleme NP-complete.