Gyakorlás

Algoritmuselmélet

1. konzultáció

2024.

1. Alkalmas c konstans és n_0 küszöbérték megadásával lássuk be, hogy $f(n) \in O\big(g(n)\big)$ teljesül az alábbi függvényekre.
$$f(n) = 2023n^2 \cdot \log n - 27\sqrt{n}, \qquad g(n) = \frac{1}{10^{10}} \cdot n^3 + n \cdot \log n$$

2. Tudjuk, hogy $f(n)$ egy nemnegatív, monoton növekvő függvény, melyre teljesül, hogy $f(n) \in O(n^2)$. Következik-e ebből, hogy $f\big(n^2 f(n) + 3f(n) + 5\big) \in O(n^4)$?

3. Bizonyítsuk be, hogy $\sqrt{n}^{\sqrt{n}} \in O(2^n)$.

4. Az $A[0:n]$ tömbre teljesül, hogy minden $2 \leq i \leq n-2$ esetén $A[i-2] < A[i] < A[i+2]$. Adjon a tömb rendezésére egy $O(n)$ összehasonlítást használó algoritmust.

5. Az $a_1, \ldots, a_n$ természetes számokról tudjuk, hogy 10 darab kivételével teljesül, hogy $10 \leq a_i \leq 10n$; a kivételes 10 természetes szám bármekkora lehet. Adjunk $O(n)$ futásidejű algoritmust, ami növekvően rendezi az összes számot.

6. Egy tányérra n különböző átmérőjű, kör alakú palacsinta van felhalmozva véletlenszerű sorrendben. Szeretnénk a palacsintákat alulról felfelé növekvő sorrendbe rendezni. Ehhez egyféle műveletet használhatunk: egy lapáttal benyúlunk bármely palacsinta alá és a felette levő kupacot fejjel lefelé fordítva visszatesszük a kupacra. Adjunk olyan algoritmust, ami minden esetben legfeljebb $2n-3$ művelettel elvégzi a feladatot.

7. Éllistájával adott egy irányított G gráf, melynek minden csúcsa színes: piros, fehér vagy zöld színű. Adott a gráfban egy A csúcs, ami piros és egy B csúcs, ami zöld. Adjunk $O(n+m)$ lépésszámú algoritmust, ami megtalálja a legkevesebb élből álló olyan utat A-ból B-be, amiben az első néhány csúcs piros, majd néhány (legalább egy) fehér csúcs után csupa zöld csúcs következik (n és m szokás szerint a gráf csúcs-, illetve élszámát jelölik).

8. Szomszédossági mátrixával adott a $V = \{v_1, v_2, \ldots, v_n\}$ csúcshalmazon a G irányított gráf. Ebben az olyan irányított körök érdekelnek minket, amelyek átmennek a v_1 csúcson és innen kezdve a kör mentén a csúcsok indexei sorrendben követik egymást; például egy k-csúcsú kör mentén a csúcsok indexe sorban $1 = i_1 < i_2 < i_3 < \ldots < i_k$. Adjunk algoritmust, ami $O(n^2)$ lépésben meghatározza, hogy mi az a legnagyobb k szám, amire G-ben van k csúcsból álló, a feltételnek megfelelő kör.

9. Adott az $a_1, a_2, \ldots, a_n$ egész számokból álló sorozat. Ebben olyan $a_{i_1}, a_{i_2}, \ldots$ részsorozatot keresünk, melynek elemei egy 5-különbségű számtani sorozatot alkotnak (azaz az értékek sorban $x, x+5, x+10, x+15, \ldots$). Adjon $O(n^2)$ lépésszámú algoritmust, ami meghatározza a leghosszabb ilyen részsorozat hosszát.

10. Adott egy pozitív egész számokból álló $a_1, a_2, \ldots, a_n$ számsorozat. Ebben egy $a_{i_1} < ai_2 < \ldots < a_{i_k}$ monoton növő részsorozat ($i_1 < i_2 < \ldots < i_k$) értéke $a_{i_1} \cdot a_{i_2} \cdot \ldots \cdot a_{i_k} + a_{i_k}$, azaz összeszorozzuk a részsorozatban szereplő összes számot és hozzáadjuk az utolsó értékét. Adjunk $O(n^2)$ lépésszámú dinamikus programozást használó algoritmust a legnagyobb értékű monoton növő részsorozat értékének megkeresésére. (Például a 2, 5, 1, 3, 10, 7, 9 sorozatban a 2, 5, 7, 9 a legjobb részsorozat, ennek értéke $2 \cdot 5 \cdot 7 \cdot 9 + 9 = 639$.)

11. Egy vizsgadolgozatban n feladatot kell megoldani, az i-edik feladatra p_i pontot lehet kapni. A feladatokat az adott sorrendben kell megoldani, de ki lehet hagyni közülük bármennyit. Ráadásul, ha a hallgató megoldja az i-edik feladatot (és így megkapja a p_i pontot), akkor annyira elfárad, hogy ki kell hagyja a következő f_i darab feladatot. (De többet is kihagyhat. A feladatsor első néhány feladatát is kihagyhatja.) Adjunk $O(n)$ futásidejű dinamikus programozást használó algoritmust, ami a p_i, f_i egész számok ismeretében meghatározza az elérhető összpontszám maximumát.

12. Egy középkori királyság úthálózata egy n-csúcsú irányítatlan gráffal adott (a csúcsok a városok, az élek a köztük vezető utak). Az A városból szeretnénk a B városba árut vinni, de bizonyos városok csak akkor engednek át minket a terményünkkel, ha vámot fizetünk nekik (az A és B városban nem kell vámot fizetnünk). A vám összege fix, nem függ az áru mennyiségétől, de a vám városonként más és más lehet. Adjunk algoritmust, ami a városonkénti vámok és a gráf szomszédossági mátrixának ismeretében $O(n^2)$ lépésben meghatároz egy olyan útvonalat, amin a legkevesebb sarcot szedik be tőlünk.

13. Szomszédossági mátrixával adott n-csúcsú, élsúlyozott, irányítatlan gráfként ismerjük egy ország úthálózatát: a csomópontok a városok, az élek a közvetlen összeköttetések a városok között, az élek súlya pedig a városok közti távolságot adja meg. (Feltehetjük, hogy a távolságok egészek.) Adjunk $O(n^6)$ lépésszámú algoritmust, ami eldönti, hogy lehetséges-e úgy kiválasztani öt várost, hogy ezektől bármely más város legfeljebb 50 kilométerre van. (Ezekbe a városokba lenne érdemes hókotrókat telepíteni.)