

Adatstruktúrák, algoritmusok vizsga Feladatlap: 1

Név (nyomtatott nagybetűk)	
Név (sajátkezű aláírás)	
NEPTUN kód.....:	
Dátum.....:	
Értékelés.....:	

Elméleti kérdések		
Zárójelben az adható pontszámok. A pontszámok tovább részpontokra nem bonthatók. Max 16 pont :Értékelés: 0-4 elégtelen, 5-6 elégséges, 7-8 közepes, 9-11 jó, 12-16 jeles		
1	Definiálja a O (nagy Ordo) és a o (kis ordo) szimbólumokat (1+1)! <i>ordo</i>:1. $f(n)=O(g(n))$: $\exists c>0$ és n_0, hogy ha $n>n_0$, akkor $0 < f(n) \leq c \cdot g(n)$ <i>Kis ordo</i>: $f(n)=o(g(n))$: $A c>0 \exists n_0$, hogy ha $n>n_0$, akkor $0 < f(n) < c \cdot g(n)$	
2	Mikor mondjuk, hogy $f(n)$ függvény polinomiálisan lassabb növekvésű, mint egy n^p polinom (1)? Mikor, hogy polinomiálisan gyorsabb (1)! Azt mondjuk, hogy az $f(n)$ növekedési fv <i>polinomiálisan gyorsabban</i> nő, mint az n^p polinom ($p \geq 0$), ha létezik olyan $\varepsilon > 0$ valós szám, hogy $f(n) = \Omega(n^{p+\varepsilon})$. Azt mondjuk, hogy az $f(n)$ növekedési fv <i>polinomiálisan lassabban</i> nő, mint az n^p polinom ($p \geq 0$), ha létezik olyan $\varepsilon > 0$ valós szám, hogy $f(n) = O(n^{p-\varepsilon})$.	
3	Fogalmazza meg a két szám lineáris kombinációi halmazának jellemzési tételét (1) és bizonyítsa be a tételt (1)! Legyen M a $d^* = \text{lko}(a,b)$ egész többszöröseinek a halmaza. Állítás: $L(a,b) \equiv M$ Bővebben: az $L(a,b)$ minden eleme d^* egész többszöröse és ha egy s szám d^* egész többszöröse, akkor az s szám az a és b lineáris kombinációja is.	
4	Adja meg a két egész szám legnagyobb közös osztójának meghatározására szolgáló euklideszi algoritmus pszeudokódját (1) és az ugyanezen célt szolgáló bináris legnagyobb közös osztó algoritmus pszeudokódját (1)?	
5	Definiálja a multiplikatív inverz fogalmát (2) ! Legyen a lineáris kongruencia egyenlet $ax \equiv 1 \pmod n$ $a \in \mathbb{Z}, n \in \mathbb{Z}^+, \text{lko}(a,n)=1$ alakú (azaz a és n egyenlenek relatív prímek). Az egyenlet egyetlen alapmegoldását az a szám n szerinti multiplikatív inverzének nevezzük. Jelölése: $x = a^{-1} \pmod n$.	
6	Mennyi az n kulcs összefésülő rendezésének időigénye (1)? Bizonyítsa be az állítását (1)! $T = \Theta(n \log n)$	
7	Definiálja a (nyílt) fagráf (1) és az erdő fogalmát (1)! Erdő :Körmentes, irányítatlan gráf. (Nyílt) fagráf :Összefüggő, körmentes irányítatlan gráf.	
8	Milyen feladatot old meg a Prim algoritmus (1) és milyen a Bellman-Ford algoritmus (1)? Bellman-Ford algoritmus A Bellman-Ford algoritmus az i-edik iterációban minden csúcshoz (kivéve kezdő csúcsot), megtalálja a minimális költségű, legfeljebb i hosszúságú utat egy irányított gráfban. Negatív élköltségek is lehetségesek a gráfban. Feltétel: a gráf nem tartalmazhat negatív kört. prim algoritmus: Minimális súlyú feszítőfa meghatározása.	

Megoldandó feladatok

Zárójelben az adható pontszámok. A pontszámok tovább részpontokra nem bonthatók.

Az egyes kérdésekre adott számszerű választ írja be a kérdés sorában a jobbszélső cellába.

Max 8 pont.

Értékelés: 0-2 elégtelen, 3 elégséges, 4 közepes, 5-6 jó, 7-8 jeles

1	Határozza meg a moduláris hatványozó algoritmussal a $123^{124} \bmod 125$ értékét!							
	a. Mennyi az érték (1) ?	91						
	b. Hány négyzetre emelést kellett elvégezni az algoritmus munkája során (1) ?	7						
2	Adott két rendezett kulcssorozat: Egyik sorozat 1, 2, 3, 4, 5, 11, 12, 13 Másik sorozat 6, 7, 8, 9, 10, 14, 15, 16, 17 Fésülje össze a két sorozatot a Batchter féle összefésülés módszerével!							
	a. Hány kulcs-összehasonlítás volt összesen az eljárás során (1)?	21						
	b. Mennyi lett volna a kulcs-összehasonlítások száma, ha közönséges összefésülést alkalmaztunk volna (1)?	13						
3	Az 1, 2, 3, 4, 5, 6, 7, 8, 9 kulcsokból hozzon létre egy bináris kupacot (maximum kupac) üres kupacból kiindulva az elemek sorrendjének megfelelő egyenkénti beszúrással!							
	a. Hány kulcs-összehasonlítás volt az eljárás során (1)?	16						
	b. Eltávolítva a maximális elemet a kupacból a KIVESZ_MAX eljárással hányszor aktivizálódott összesen a KUPACOL eljárás (1)?	4						
4	Adott egy hálózat (súlyozott gráf) az 1, 2, 3, 4, 5, 6 csúcsokkal és az alábbi élekkel: <table border="1"><tr><td>1 → 2(1), 3(2), 4(3);</td><td>4 → 6(9);</td></tr><tr><td>2 → 3,(4) 5(6);</td><td>5 → 6(10);</td></tr><tr><td>3 → 4(5), 5(7), 6(8);</td><td></td></tr></table> A Kruskal algoritmussal keressen egy minimális feszítőfát!		1 → 2(1), 3(2), 4(3);	4 → 6(9);	2 → 3,(4) 5(6);	5 → 6(10);	3 → 4(5), 5(7), 6(8);	
1 → 2(1), 3(2), 4(3);	4 → 6(9);							
2 → 3,(4) 5(6);	5 → 6(10);							
3 → 4(5), 5(7), 6(8);								
	a. Mennyi a fa súlya (1)?	20						
	b. Az algoritmusnak megfelelően minimálisan hány él megvizsgálása után nem kerül sor további EGYESIT eljáráshívásra (1)?	8						