

Név:	Neptun kód:	Pontszám:
------	-------------	-----------

**1. Feladat (6 pont)**

Adott az alábbi bemenő sorozat ( $A$  tömb): 3, 5, 8, 3, 2, 2, 7, 8, 9, 2. A sorozat elemei 0 és 9 közötti egész számok. Rendezzük a sorozatot leszámláló rendezéssel egy  $B$  tömbbe egy  $C[0 \dots 9]$  segédtömb segítségével.

**2. Feladat (8 pont)**

Valósítsuk meg a hash-tábla törlő műveletét ( $Remove(H, k)$ , ahol  $H = (n, t[0 \dots M - 1])$ ,  $n$  az elemek száma,  $t$  egy  $M$  méretű tömb,  $k$  a törlendő kulcsérték,  $\emptyset$  jelöli az üres mezőt,  $X$  a törölt mezőt) arra az esetre, amikor a hash-függvény osztómódszert használ, az ütközésfeloldásra pedig lineáris próbálást alkalmazunk.

**3. Feladat (12 pont)**

Valósítsuk meg a  $Transform(Adj[1 \dots n], C[1 \dots n, 1 \dots n])$  algoritmust, amely az első paraméterben megadott éllistásan ábrázolt irányított, súlyozott, egyszerű gráfot átalakítja csúcsmátrixos reprezentációra (a második paraméterbe). Adjuk meg és indokoljuk az algoritmus műveletigényét.

**4. Feladat (8 pont)**

Készíts algoritmust, amely egy csúcsmátrix segítségével ábrázolt egyszerű, súlyozott gráfra megadja a legrövidebb utak hosszát minden lehetséges csúcspárra. Az algoritmus visszatérési értékben jelezze, hogy az eredménye helytálló-e (azaz nem volt negatív összköltségű kör a gráfban, ami elronthatja az eredményt). Az algoritmus  $\Theta(n^3)$  műveletigényét használhat fel.

**5. Feladat (12 pont)**

Valósítsd meg a Prim algoritmust hatékony reprezentációval sűrű gráfok esetén (ahol az élek száma nagyságrendileg megegyezik a csúcsok számával). Adjuk meg az algoritmus műveletigényét, és indokoljuk (a műveletigény segítségével), hogy a reprezentáció miért a leghatékonyabb sűrű gráfokra, és miért nem hatékony ritka gráfok esetén.

**6. Feladat**

- Adott az „aassaadabasddfabassssb” inputsorozat. Készítsd el az inputsorozatnak megfelelő Huffman-kódfát, illetve állítsd elő a kódtáblát. **(4 pont)**
- Adott a „aaaabababababa#” inputsorozat. Tömörítsd a sorozatot (lezárva #-kal) LZW kódolás segítségével. Építsd fel a kódtáblát az egyes karakterek beolvasását követően. Tegyük fel, hogy az alap kódtábla az angol ABC 26 betűje, valamint a vége jel: (# = 0) (a = 1), (b = 2). **(8 pont)**

Mindkét esetben számítsd ki a tömörítési arányt (figyelembe véve, hogy eredetileg 27 lehetséges karakternek megfelelő hosszon írtuk le a szöveget). **(2 pont)**

**Jó munkát!**