

Adatstruktúrák és Algoritmusok ZH

2024.10.15.

Rendelkezésre álló idő: 60 perc Semmilyen segédeszköz nem használható. A 31 pontból 16 elérése szükséges.

0. Feladat: [I/H] - 7p

Mely állítások igazak, hamisak az alábbiak közül?

- A selection sort egy már (jó irányban) rendezett tömbön $O(n)$ idő alatt fut le.
- Az insertion sort minden esetben $O(n^2)$ idő alatt fut le.
- (Nem rendezett) láncolt listában megtalálni egy elemet gyorsabb, mint kitörölni.
- Dinamikus tömbben az elemek sorrendjének megfordítása $O(n)$ időben lehetséges.
- Stabílnak azokat a rendező algoritmusokat nevezzük, amelyek egy már rendezett tömbből $O(n)$ idő alatt megállapítják, hogy jól rendezett.
- Összehasonlításra alapuló rendező algoritmusok esetében a legjobb best case futási idő $O(n)$.
- Bináris keresőfában a következő legkisebb elem átlagosan $O(\log n)$ időben megadható.

1. Feladat: $O(?)$ - 4p

Milyen aszimptotikus futási ideje van (legrosszabb esetben) az alábbi kódrészletnek `n=len(l)` függvényében? A választ indokolja, és adjon konstrukciót olyan bemenetre, ahol ez előfordul.

```
l2 = l[:]
for idx in range(int(len(l)*0.5)):
    if l[idx] > l[idx+1]:
        l2.insert(l[idx],0)
```

2. Feladat: HEAP - 4p

Adott egy MAX-HEAP a `[26, 18, 19, 11, 3, 7]` elemekkel. Hogy néz ki a kupac tartalma az alábbi műveletek után? (tömb és fa):

1. `push(52)`
2. `push(16)`
3. `push(90)`
4. `pop_max()`

3. Feladat: Sort - 6p

Rendezze az alábbi tömböt quick sorttal: `[11, 8, 6, 4, 9, 23, 2, 7, 55, -9, 5]`. A rendezés köztes lépései látszódnak. Pivot elemnek mindig az első elem legyen választva, a partíciókon belül az elemek sorrendje legyen ugyanaz, mint ami a felosztás előtti.

4. Feladat: Sort 2 - 5p

Írja le a counting sort működési elvét, aszimptotikus futási idejét, valamint használatának feltételeit.

5. Feladat: BFS -5p

Adott egy üres bináris keresőfa, mi a fa állapota (ábrákkal) az alábbi lépések után?

1. `push(6)`
2. `push(-6)`
3. `push(9)`
4. `push(16)`
5. `push(61)`
6. `push(8)`
7. `push(63)`
8. `push(7)`
9. `push(5)`
10. `push(16)`
11. `delete(16)`
12. `push(15)`
13. `delete(9)`
14. `delete(8)`