

1. kérdés

2 / 2 pont

Melyek igaz állítások a **tanult quicksort** algoritmussal kapcsolatban?

Helyes!

☐ Előfordulhat, hogy a particionálást követően a tengely utáni rész üres, azaz nulla méretű, azonban a tengely előtti rész soha nem lehet üres.

Helyes!

☐ A quicksort legkedvezőbb esetben lineáris időben rendez.

Helyes!

☒ A tengely kiválasztása és a részekre bontás mindig lineáris időben befejeződik.

Helyes!

☒ A quicksort "oszd meg és uralkodj" elvű algoritmus.

☐ A quicksort stabil rendezés.

☐ A quicksort tetszőleges bemeneten gyorsabb, mint a beszűrő rendezés.

Helyes!

☒ A particionálás egy-elemű résztömbre nem hajtódik végre.

Helyes!

☒ A quicksort particionáló eljárása közben a tengellyel egyenlők a tengely elé, és mögé is kerülhetnek.

2. kérdés

0 / 1 pont

A "p" pointer egy egyszerű lista (S1L) egy *tetszőleges* elemére mutat. Az "r" pointer egy E1 típusú elemre mutat. Be szeretnénk fűzni a listába az r című elemet a p című elem mögé. Válassza ki a helyes megoldásokat az alábbiakból.

Megadott válasz

☒ (*r).next:=p; (*p).next:=r ;

Helyes!

☐ p→next:=r; r→next:=p→next;

Helyes!

☒ r→next:=p→next; p→next:=r;

Helyes válasz

☐ (*r)→next:=(*p).next; p→next:=r;

☐ (*r).next:=p→next; (*p).next:=r;

3. kérdés

1 / 1 pont

Melyek igazak a tanult egyszerű listára (S1L) az alábbi állítások közül?

Helyes!

☒ p→next:=p→next→next; hatása: kifűzi a p utáni elemet a listából. (Tudjuk, hogy p ≠ 0 és p→next ≠ 0 .)

☐ A lista bármely i-edik elemének lekérdezése konstans időben megvalósítható. (1≤i≤n, ahol n a lista elemszáma.)

☐ Ha a listából törölni akarunk egy tetszőleges elemet, mindig szükség van a törlendő elem előtti elem címére.

Helyes!	<input checked="" type="checkbox"/> A lista utolsó elemének elérése $\theta(n)$, ahol n a lista elemszáma. (Csak a lista első elemének címét ismerjük.)
Helyes!	<input checked="" type="checkbox"/> Ha a lista egy p című eleme elé szeretnénk befűzni egy új elemet, akkor annak műveletigénye $O(n)$, ahol n a lista elemszáma. (A p című elemen kívül csak a lista első elemének címét ismerjük.)

4. kérdés2.5 / 3 pont

Rendezze a **tanult** gyorsrendezéssel (quicksort) a $\langle 8,7,13,6,11,3,1,9,4 \rangle$ egész számokat tartalmazó tömböt!

Feltesszük, hogy a particionálás minden esetben az aktuális résztömb **első** elemét választja tengelynek.

Adja meg sorban a $\text{partition}(A; p; r)$ segédfüggvény hívásai által kiszámolt résztömböket, az elemeik felsorolásával, a tengelyt a '+' előjellel különböztetve meg. Az elemeket vesszővel válassza el, szóköz ne legyen a szövegben. Például: 4,2,+5,8

4,7,6,3,1,+8,11,9

1,3,+4,7,6

+1,3

6,+7

9,+11,13

Hány kulcs összehasonlítás történt a rendezés során: 12

1. válasz:

4,7,6,3,1,+8,11,9,13

2. válasz:

1,3,+4,7,6

3. válasz:

+1,3

4. válasz:

6,+7

5. válasz:

9,+11,13

6. válasz:

12

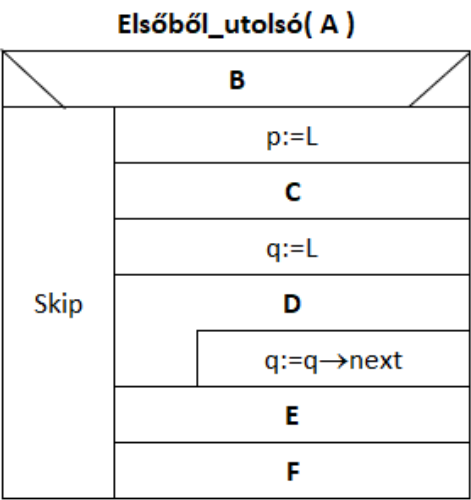
Megadott válasz

Helyes válasz

16

5. kérdés3 / 3 pont

L egy egyirányú, fejelem nélküli (S1L) lista első elemére mutat. A lista első elemét át szeretnénk fűzni az utolsó elem után. Ha a lista üres, vagy egy-elemű, ne történjen semmi. Válassza ki a legördülő listákból az alábbi struktogramba illő helyes lépéseket.



A:

[Kiválaszt] ▼

B: L= 0 v L→next=0

C:

[Kiválaszt] ▼

D:

[Kiválaszt] ▼

E:

[Kiválaszt] ▼

F:

[Kiválaszt] ▼

1. válasz:

Helyes!

&L : E1*

2. válasz:

Helyes!

L= 0 v L→next=0

3. válasz:

Helyes!

L := L→next

4. válasz:

Helyes!

q→next ≠ 0

5. válasz:

Helyes!

q→next := p

6. válasz:

Helyes!

p→next := 0