Řazení III cvičení

1.

- Pomocí Radix sort-u řadíme pole n řetězců, každý řetězec má kladnou délku k. Asymptotická složitosrt tohoto řazení je

a) $\Theta(k^2)$
b) $\Theta(n^2)$
c) $\Theta(k \cdot n)$
d) $\Theta(n \cdot \mathrm{loJ}(n))$
e) $\Theta(k \cdot n \cdot \mathrm{loJ}(n))$

2a.

- Následující posloupnost řetězců je nutno seřadit pomocí Radix Sortu (přihrádkového řazení). Proveďte první průchod (z celkových čtyř průchodů) algoritmu danými daty a napište, jak budou po tomto prvním průchodu seřazena.

IIYI PIYY YIII YPPP YYYI PYPP PIPI PPYI

2b.

- Následující posloupnost řetězců je nutno seřadit pomocí Radix Sortu (přihrádkového řazení). Proveďte první průchod (z celkových čtyř průchodů) algoritmu danými daty a napište, jak budou po tomto prvním průchodu seřazena.

JKKJ KEEJ JKJJ KJKK KJEE KEEJ EJEE JEEJ

3.

- Pole A obsahuje téměř seřazené řetězce (např. z 99% seřazené), pole B obsahuje řetězce stejné délky, ale zcela neseřazené. Radix sort seřadí asymptoticky

a) A rychleji než B
b) B rychleji než A
c) A stejně rychle jako B, použije více paměti pro řazení A
d) A stejně rychle jako B, použije více paměti pro řazení B
e) A stejně rychle jako B a použití paměti bude stejné

4a.

- Radix sort řadí dané pole řetězců. Napište, jak budou zaplněna jednotlivá pomocná pole (z, k, d, podle přednášky) po prvním průchodu algoritmu tj, po seřazení podle posledního znaku.

dda, bab, ddc, aaa, bcd, dbc, bbb, add, ccd, dab, bbc

4b.

- Radix sort řadí dané pole řetězců. Napište, jak budou zaplněna jednotlivá pomocná pole (z, k, d, podle přednášky) po prvním průchodu algoritmu tj, po seřazení podle posledního znaku.

dad, caa, cad, aac, bca, dbc, bbd, ddc, cda, dac, bbc

5.

- Během provádění Radix sortu známe obsah polí z1, k1, d1 (viz přednáška). Máme určit, podle kterého znaku právě řazení probíhá. To určit

a) lze, index tohoto znaku je uložen v k1[0]

b) lze, index tohoto znaku je uložen v d1[0]

c) lze, index tohoto znaku je uložen v z1[0]

d) nelze, protože algoritmus tuto informaci v d1 soustavně přepisuje

e) nelze, pole z1, k1, d1 tuto informaci neregistrují

6.

- Během provádění Radix sortu známe obsah polí z1, k1, d1 (viz přednáška). Máme určit, zda v tomto okamžiku již byl dokončen jeden z průchodů algoritmu vstupními daty . To určit

a) nelze, neznáme pořadí prvků ve vstupním poli

b) nelze, některé prvky v poli d1 mohou být stejné

c) nelze, některé prvky v polích z1, k1 mohou být stejné

d) lze, sečtením délek všech seznamů uložených v poli d1

e) lze, sečtením rozdílů odpovídajících si prvků v k1 a z1

7a.

- Radix sort právě řadí podle 3 znaku od konce a ještě průchod nedokončil. Jak se bude postupně měnit obsah polí po zařazení každého z dalších řetězců?

  abbac (9), **aaaaa** (4), **bbbac** (6),

  (čísla) uvádějí pozici řetězce ve vstupním poli.

  Aktuální obsah polí je

z1

a	b	c
3	10	5

k1

a	b	c
1	2	5

d1

1	2	3	4	5	6	7	8	9	10
0	0	8	0	0	0	2	1	0	7

7b.

- Radix sort právě řadí podle 4 znaku od konce a ještě průchod nedokončil. Jak se bude postupně měnit obsah polí po zařazení každého z dalších řetězců?

baaab (11), **ccaba** (4), **ababa** (6), **babab** (5), **bbbbc** (10)

(čísla) uvádějí pozici řetězce ve vstupním poli.

Aktuální obsah polí je

z1

a	b	c
8	0	3

k1

a	b	c
1	0	2

d1

1	2	3	4	5	6	7	8	9	10	11
0	0	2	0	0	0	9	7	1	0	0

8.

- Každé kladné číslo typu **int** lze interpretovat jako posloupnost čtyř znaků reprezentujících jeho zápis v soustavě o základu 256.

- Jak je nutno modifikovat Radix sort, aby mohl řadit pole takto interpretovaných celých čísel?
- Bude to časově/paměťově výhodné? Pro jak veliká pole?

9.

- Uvažujme pole obsahující 1000 navzájem různých čísel v pohyblivé řádové čárce. Counting sort se pro toto pole

a) hodí, protože toto řazení má lineární složitost
b) hodí, protože toto řazení má sublineární složitost
c) hodí, protože čísla lze převést na řetězce
d) nehodí, protože čísla v poli nemusí být celá
e) nehodí, protože čísla jsou navzájem různá

10.

- Counting sortem řadíme pole čísel:

8 14 14 7 11 11 6 3 12 11 2 12 14 9 8

- Jaký bude počáteční obsah pole četností pro tato data?

11a.

- Counting sortem řadíme pole čísel:

  50 88 87 87 93 87 23 53 70 89 53 62

  Jaký bude nejmenší a největší index v poli četností?

a) 50 62
b) 0 62
c) 0 93
d) 23 93
e) 50 93

11b.

- Counting sortem řadíme pole čísel:

  56 54 20 13 44 75 84 39 31 34 68

  Jaký bude nejmenší a největší index v poli četností?

a) 20 56
b) 20 84
c) 1 56
d) 56 84
e) 56 68

12a.

- Řadíme 14 celých čísel. Těsně předtím, než se začne plnit výstupní pole v Counting sortu, je obsah původního pole četností následující (slabě psaná čísla jsou indexy):

10	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*
1	**3**	**4**	**8**	**10**	**12**	**12**	**12**	**13**	**14**

- Napište, jak vypadá výsledné uspořádané pole, za předpokladu, že všechna pole začínají indexem 1.

12b.

- Řadíme 15 celých čísel. Těsně předtím, než se začne plnit výstupní pole v Counting sortu, je obsah původního pole četností následující (slabě psaná čísla jsou indexy):

10	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*
0	**1**	**2**	**3**	**7**	**9**	**9**	**13**	**14**	**15**

- Napište, jak vypadá výsledné uspořádané pole, za předpokladu, že všechna pole začínají indexem 1.

13.

- Na začátku Counting sortu je v poli četností uložena právě četnost výskytu každé hodnoty ve vstupním poli. V průběhu řazení se obsah pole četností průběžně mění.
- Rozhodněte, zda je možno po dokončení celého řazení rekonstruovat z modifikovaného pole četností původní (a ovšem i výsledné) četnosti jednotlivých řazených hodnot.