Name: Guodong Chen

Zid: z5224156

Q1(p1, p2, p$)

1 answer ← ∅;

2 while p1 != nil ∧ p2 != nil do

3 if docID(p1) == docID(p2) then

4 l ← [ ];

5 pp1 ← positions(p1); pp2 ← positions(p2);

6 while pp1 != nil do

7 pp$ ← skipTo(p$, docID(p1), pp1);

8 while pp2 != nil do

9 if pos(pp2) < pos(pp$) then

10 add(l, pos(pp2));

11 else

12 break;

13 pp2 ← next(pp2);

14 while l != [ ] ∧ pos(pp1) > l[1] do

15 delete(l[1]);

16 for each ps ∈ l do

17 answer ← answer ∪ [docID(p1), pos(pp1), ps];

18 pp1 ← next(pp1);

19 p1 ← next(p1); p2 ← next(p2);

20 else

21 if docID(p1) < docID(p2) then

22 p1 ← next(p1);

1. else

24 p2 ← next(p2);

25 return answer;

Q2

(1)

As we have t sub-indexes in no merge, so there are t batch of documents.

If we use logarithmic merge strategy for t batch of documents, then we have

|  |  |  |
| --- | --- | --- |
| t-th of documents | Sub-indexes(Ik) | Number of sub-indexes |
| 0 | 0 | 0 |
| 1st | I0 | 1 |
| 2nd | I0 + I0 = I1 | 1 |
| 3rd | I1 + I0 | 2 |
| 4th | I1 + I1 = I2 | 1 |
| 5th | I2 + I0 | 2 |
| 6th | I2 + I0 + I0 = I2 + I1 | 2 |
| 7th | I2 + I1 + I0 | 3 |
| 8th | I2 + I1 + I0 + I0 = I3 | 1 |
| ... | ... | ... |
| t-1-th | ... | Log2t |
| t-th | lLog2t | 1 |

From above table, we find if the logarithmic merge strategy is used, it will result in at most ⌈log2t⌉ sub-indexes.

(2)

As we have *t* sub-indexes (each of *M* pages), thus we have t\*M entry, each entry I/O cost is log2t, then the total I/O cost of the logarithmic merge is *O*(*t · M ·* log2 *t*).

Q3

|  |  |  |  |
| --- | --- | --- | --- |
| VB code | 01000101  11110001 | 01110000  00110000  11110110 | 11011 |
| gaps |  | 21+22+24+25+26+211+212+218+219  +220=1841270 | 20+21+23=11 |
| docIDs | 20+24+25+26+27+29+213=8945 | 8945+1841270=1850215 | 1850215+11=1850226 |

Q4

As function pickTerm() choose term is based on maximal idf, which mean minimal df, if pickTerm() choose a term that the term’s DID include pivot and the df of pterm and the chosen term are lower than other term’s df, then pickTerm() will always choose the two term which end up in a infinite loop.

|  |  |  |  |
| --- | --- | --- | --- |
|  | A | B | C |
| UB | 3 | 4 | 5 |
| List | <1,3> | <1,4> | <1,4> |
| <2,1> | <3,3> | <3,3> |
| <3,1> |  |  |

