1.3 Processing Bool query

e1. Brutus And Calpurnia

· posting intersection; posting merge

ex 1.4 ($A \wedge \neg B$; $A \vee \neg B$) complexity

1.5 $(A \vee B) \wedge \neg (C \vee D)$.

1.6 a. ↳ DNF $(\neg A \wedge \neg B) \vee (C \vee D))$

b. which is more efficient?

c. analyze efficiency

1.7 $(a \vee b) \wedge (c \vee d) \wedge (e \vee f)$

recommend a query process.

a: 200k ; b: 87k ; c 108K ; d 270k , e 46k f 317k

(Ref to 1.3(1.5)).

↳ $(a \vee b) \wedge (e \vee f) \wedge (c \vee d)$

1.8 $a \wedge b \vee c$. ☆

discuss $\underbrace{\text{freq(C)} \longleftrightarrow \text{order of query processing}}_{\text{opt.}}$

([freq(a), freq(b), N-freq(C)].sorted()

. generate the posting list for "Not countryman".

. order these terms by their updated p~list.)

1.9. Not guaranteed | It's optimal for worst case.

example: doclist: ["c", "c", "c", "a b", "a b"].

② query: $a \wedge b \wedge c$;