Lecture :- Contest 3 discussion

Agenda

- Similar elements
- Simple recursion
- Detective and unbanned words.

P&p > 90 %.

Attendance > 80 %.

Q. Given a string[n] ⟶ conversations

string[n] ⟶ banned words

Return the most frequently word in consvation that is not in the banned word.

If there are multiple words with highest frequency, return the lexographically smallest one

Example:

conversations[] = [aa, bb, bb, aa, aa] ans = bb

banned = [aa, cc]

aa - 3

bb - 2

conv = [xy, pq, d] pq Any xy - 1 ✓

banned = [e, d]

pq - 1

d - 1

Observations:

1. freq of each word in conservation array.
2. Check whether a word is inside the banned array or not?

↓

Hash Set

3. If freq are same, go for lexographical logic.

eg: conservation[ ] = [ab, cd, ef, ab, cd, bc, fc]
(0) (1) (2) (3) (4) (5) (6)

banned = [ab, d, f, lm, qb]

bannedset[ ] = [ab, d, f, lm, qb], freqmap[ ] = [ ]

ans = " "

conservation[ ] = [ab, cd, ef, ab, cd, bc, fc]

i = 0 — bannedset contains (ab)

|

cannot be my answer.

freqmap = [ ]

ans = " "

---

i = 1 bannedset contains(cd)

↓

NO

freqmap[ ] = [cd : 1]

ans = cd.

---

i = 2 bannedset contains(ef)

↓

NO

freqmap[ ] = [cd : 1, ef : 1]

ans = cd.

compare freq of currword with your ans

freq(ans) = 1
(cd)

freq(curr-word) = 1
(ef)

ans.compareto(curr-word) < 0
[cd] [ef]

ans is lexographically smaller

i = 3 ab

bannedset: contains (ab)

|

cannot be my answer.

freqmap = [cd:1, ef:1]

ans = cd

---

i = 4 conservation[] = [ab(0), cd(1), ef(2), ab(3), cd(4), bc(5), bc(6)]

bannedset: contains (cd)

↓ no

freqmap = [cd:2, ef:1]

freq(ans(cd)) = 2.

freq(currword(cd)) = 2

ans = cd

cd.compare(cd) == 0

ans curr-word

---

i = 5 [bc]

bannedset: contains (bc)

↓ no

freqmap[] = [cd:2 ef:1, bc:1]

freq(ans(cd)) = 2

freq(curr-word(bc)) = 1

ans = cd.

---

i = 6 [bc]

bannedset: contains (bc)

↓ no

freqmap[] = [cd:2 ef:1, bc:2]

freq(ans(cd)) = 2

freq(curr-word(bc)) = 2

ans = curr-word = bc

ans.compareTo(curr-word) > 0

cd bc

---

abc, abd

abc.compareTo(abd) < 0

abd.compareTo(abc) > 0

Qu 2.

243 Ans

foo(3, 5)

bar(3, foo(3, 4)) 81 3 * 81 = 243

bar(3, foo(3, 3)) 27 3 * 27 = 81

bar(3, foo(3, 2)) 9 3 * 9 = 27

bar(3, foo(3, 1)) 3 = 3 * 3 = 9

bar(3, foo(3, 0)) y == 0 (1)

bar(3, 1) = x * y = 3 * 1 = 3

```
int bar(x, y) {
    if (y == 0) {
        return 0;
    }
    return x + bar(x, y-1);
}

int foo(x, y) {
    if (y == 0) {
        return 1;
    }
    return bar(x, foo(x, y-1));
}

main() {
    print(foo(3, 5));
}
```

```
int bar(x, y) {
    if (y == 0) {
        return 0;
    }
    return x + bar(x, y-1);
}
```

bar(x, y) = x + bar(x, y-1)

= (x + x) + bar(x, y-2)

= (x + x + x) + bar(x, y-3)

= (x + x + x + x) + bar(x, y-4)

= x + x + x + x --- (y times)

bar(x, 0)

0

bar(x, y) = x + x + x + x --- (y times) = x * y