Hash Map: Classical Problems

Course on Basic Data Structures (C++)

Pulkit Chhabra • Lesson 16 • Feb 16, 2021

Q. Given 2 arrays x & y. find no. of pairs of distinct integers (i, j) s.t. $x[i] - x[j] = y[i] - y[j]$.

	0	1	2
Eg. $x =$	3	1	5
$y =$	9	2	11

$(0, 2)$

$(2, 0)$

Ans = 2

Sol:-

$$x[i] - x[j] = y[i] - y[j]$$

$$\Rightarrow \quad x[i] - y[i] = x[j] - y[j]$$

```
umm f;

for (i=0; i<n; ++i)
    f[x[i] - y[i]]++;

ans = 0;
for (auto p : f)
{  ll n = p.second;
   ans += n*(n-1);
}
```

2 distinct
& order matters

$n * (n-1)$

$5 \to x_i - y_i = 5$

for i: 1, 2, 5, 8, 10, 11

n = 6

$\underline{n} * \underline{(n-1)}$

We have a list of strings. Find the number of pairs (i,j) s.t. $0 \le i < j \le n-1$ & $W[i] + W[j]$ can be shuffled to form a palindrome.

$W = [ab, bcdede, cbd]$

$(0, 1)$ abbcdede $\Rightarrow$ No

$(0, 2)$ abcbd $\Rightarrow$ No

$(1, 2)$ bcdedecbd

$\Rightarrow$ dbcedecde

Ans = 1

1) Order doesn't matter

2) Parities of frequencies of diff. characters matter

3) In $w[i] + w[j]$, number of characters with odd freq. should be less than 2.

'a', 'b', 'c', 'd' . . . — . . 'z'

$\{ 0, 1, 1, 0, 1 . — . . \}$

Example of how to hash

$0 * 2^0 + 1 * 2^1 + 1 * 2^2 + 0 * 2^3 + 1 * 2^4 \cdots$

$\Rightarrow hash(s_1 + s_2) = hash(s_1) + hash(s_2)$

s_1 , s_2

$hash(s_1)$

$hash(s_2)$

b_1	b_2	$b_1 \oplus b_2$
0	0	0
0	1	1
1	0	1
1	1	0

$0, 2^0, 2, 4, 8, 16 \ldots\ldots 2^{25}$

Possible values of xor

⇒ Min_time wala remove

2) value of a key

2) Update time of a particular to most recent time

2) remove a particular key

key → time

key → value

time → key

Sol. 1 :-

Get → $O(1)$

Put → $O(N)$

Sol. 2 :→

Get → $O(\log N)$

Put → $O(\log N)$

Required :

Get → $O(1)$

Put → $O(1)$