? Problem Summary: Water Filling

Chef has 3 water bottles. Each can either be full (`1`) or empty (`0`).

- If **at least 2 bottles are empty**, Chef fills them — output: `"Water filling time"`.
- If **at most 1 bottle is empty**, Chef waits — output: `"Not now"`.

Input:

- First line: Integer `T` — number of test cases.
- Next `T` lines: Each line has 3 integers (`B1`, `B2`, `B3`) — 1 for full, 0 for empty.

Output:

- For each test case, output:
  - `"Water filling time"` → if **2 or more bottles are empty**.
  - `"Not now"` → otherwise.

Logic:

Count how many zeros (empty bottles).
If count ≥ 2 → `"Water filling time"`
Else → `"Not now"`

Example:

Input:

```
5
0 0 0
1 1 1
1 1 0
0 1 0
0 1 1
```

Output:

```
Water filling time
Not now
Not now
Water filling time
Not now
```

My Solve Approach :

Used my DLD logic

Let's Try All 8 Cases:

x	y	z	Empty Bottles	Your `result`	Expected Output
0	0	0	3	0	Water filling time ✅
0	0	1	2	0	Water filling time ✅
0	1	0	2	0	Water filling time ✅
0	1	1	1	1	Not now ✅
1	0	0	2	0	Water filling time ✅
1	0	1	1	1	Not now ✅
1	1	0	1	1	Not now ✅
1	1	1	0	1	Not now ✅

cpp Copy Edit

```cpp
int result = y * (x ^ z) + (x * z);
```