All repeating except two in C++

```
#include <iostream>
#include <vector>
using namespace std;
void solution(vector<int>& arr) {
  int xxory = 0;
  for(int val : arr) {
     xxory = xxory ^ val;
  int rsbm = xxory & -xxory;
  int x = 0;
  int y = 0;
  for(int val : arr) {
     if((val \& rsbm) == 0) {
       x = x ^ val;
     } else {
       y = y ^ val;
  }
  if(x < y) {
     cout \ll x \ll endl;
     cout \ll y \ll endl;
  } else {
     cout \ll y \ll endl;
     cout \ll x \ll endl;
  }
}
int main() {
  vector<int> arr = \{2, 2, 3, 3, 6, 6, 9, 1\};
  solution(arr);
  return 0;
}
```

Given:

 $arr = \{2, 2, 3, 3, 6, 6, 9, 1\}$

Pairs: 2, 2, 3, 3, 6, 6

Unique: $9, 1 \leftarrow$ these are the ones we need to find.

Q Step-by-step Dry Run:

Step 1: Find xxory = XOR of all elements

Iteration	val	xxory (XOR so far)
init		0
1	2	0 ^ 2 = 2
2	2	2 ^ 2 = 0
3	3	0 ^ 3 = 3
4	3	3 ^ 3 = 0
5	6	0 ^ 6 = 6
6	6	6 ^ 6 = 0
7	9	0 ^ 9 = 9
8	1	9 ^ 1 = 8

So, xxory = 8 (binary: 1000)

Step 2: Find the rightmost set bit of xxory

rsbm = xxory & -xxory = 8 & -8 = 8

Rightmost set bit is in position 4 (binary 1000)

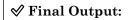
Step 3: Divide numbers into two groups based on that bit

Group 1: (val & rsbm) == 0 Group 2: (val & rsbm) != 0

val	Binary	& rsbm (1000)	Group	x or y result
2	0010	0000	x	$x = 0 ^2 = 2$
2	0010	0000	x	$x = 2 ^2 = 0$
3	0011	0000	x	$x = 0 ^ 3 = 3$
3	0011	0000	x	$x = 3 ^ 3 = 0$
6	0110	0000	x	$x = 0 ^6 = 6$
6	0110	0000	x	$x = 6 ^6 = 0$
9	1001	1000	у	$y = 0 ^9 = 9$
1	0001	0000	x	$x = 0 ^ 1 = 1$

So final values:

- $\bullet \quad \mathbf{x} = 1$
- y = 9



cout << x << endl; cout << y << endl;</pre>

Since 1 < 9, the output is:

1

Summary Table:

Element	Group	x/y update
2	X	$x \stackrel{\wedge}{=} 2 \rightarrow 0$
3	x	$x \stackrel{\wedge}{=} 3 \rightarrow 0$
6	X	$x \stackrel{\wedge}{=} 6 \rightarrow 0$
1	X	x ^= 1 → 1
9	У	$y = 9 \rightarrow 9$

1 9