

1. `string binnum = bitset<32>(n).to_string();`
`Int num = stoi(binnum, 0, 2);` //3 rd argument specifies base
2. Always pass 2 D vector by reference ex `fun(vector<vector<int>> & v).`
3. Always take long long int instead of int
Bcaz int multiplaction overflow is possible
4. Remove consecutive duplicate elements in array
`it = unique(v.begin(), v.end());`
`// resize the new vector`
`v.resize(distance(v.begin(), it));`
5. Mod with negative number will return negative number
`x = ((x % k) + k) % k ;`
6. Do case `vector.size()` to `int`. Since it returns `size_t` and will cause a problem with empty vector.
7. When we want to traverse in all 8 dire from given point
`static int rowNbr[] = { -1, -1, -1, 0, 0, 1, 1, 1 };`
`static int colNbr[] = { -1, 0, 1, -1, 1, -1, 0, 1 };`
`bool isSafe(row,col)`
`return (row >= 0) && (row < ROW) && (col >= 0) &&`
`(col < COL)`

`for (int k = 0; k < 8; ++k)`
`if (isSafe(M, row + rowNbr[k], col + colNbr[k]))`