Vanya and fence:

* Easy mathematics.

Anton and Danik:

* It is said to a int n and a string that has n numbered elements.
* So loop doesn’t help to take n numbered elements.
* N numbered character can be taken. But actually is of no use, cause you can’t make them behave like int arrays.
* So result is string and int arrays aren’t same.

Bear and big bro:

* Where we are supposed to generate loop without test cases, there while loop can be generated.

Team:

* Here we used while instead of a nested for. No matter what way we follow in case of for the result is the summation of each array elements per line.
* But in case of while loop each line’s work is done first then it goes for the next lines job.

Beautiful matrix:

* Here what we did is , took a 2d array and just did simple math with the positions.

Gravity flip:

* Here firstly we tried to check if a[i]>a[j] and if so then we incremented a[j] and decremented a[i]. But there is always a mathematical error. Check the tests of submission. So Shakil showed a way to sort out the numbers in ascending order and then print out. That’s actually genius, but not pure math actually.

Petya and string:

* Firstly converted the uppercases to the lowercases and after that checked if a[i]>a[j] and vice versa and simple print.

Boy or girl:

* It’s tough to count if I generate a loop and check a[i]==a[j]. Cause it gives surplus counts. That we don’t need. Another thing is that we can check elements side by side. So we did is sorted out the string up to ascending order and then checked side by side, and did simple math.

Word:

* What we did here is checked how many are uppercase and lowercase. Then if the number of uppercase letters are more than the lowercase then convert the whole word into uppercase and vice versa. Did simple math here.

Word Capitalization:

* Nothing serious just convert the lowercase to uppercase.

Lucky Division:

* Simple math.

Sereja and Dima:

* Here the thing is that we will check the most left and right. Then the highest one will be selected by sereja and the next turn will be for dima.
* To take turns we set a variable and checked. A new style actually.

Magnet:

* In this case we had to do some mathematics. Didn’t take me time to solve.

Stones on the table:

* Simple stuff counting checking a[i] and a[i+1].

Police Recruitment:

* Here I took help for the logic. In this case we didn’t check for a[i] and a[i+1] or a[i] and a[j]. What we did here is checked each a[i] and counted if that’s greater than -1 and if the next a[i] is -1 we subtracted until it turned zero by -1’s occurrence. Then just calculated the -1s that didn’t had any positive number in the beginning.

Black squares:

* Here the problem wasn’t clear to me at first. It’s a simple mathematics problem.
* A[] refers the calories and s[] refers the number strips jury presses.

Night at the museum:

* Took help again.
* The logic was simple enough. Just calculation.

Games:

* This is nuts. It could have been done in a easy way with 1D array.
* I just did it with 2D array.
* Did it all by myself that’s the bummer.

Buy a shovel:

* Here we did a algorithm to just increment the counter and checked if the modulus is equal to zero or the r.

Horseshoes:

* Sorted and counted by a[i] and a[i+1] method.

Colorful Stones (simplified edition):

* There’s a new technique. That is that if I have to count anything I can set that as a array’s counter as well. Like c=0 is a variable. So t[i]==s[c] we are checking. If checked then that will count 1.

Wet shark odd even:

* Array element can’t be a large amount, so using array showed TLE.
* Used a variable and checked if n is smaller than the previous n. if so and also it’s odd then we kept it in a variable. And if the sum is odd then we subtract that n that was saved.

Fortune telling:

* Almost same as wet shark one. But as here there is no high amount of element of array, we used array to solve. Tried the other way but here we needed to check side by side element.

Helpful Math:

* Counter change and sorted.

Die Roll:

* Not efficient enough. Did elementary stuffs.

Shaass And oskols:

* Maths
* And another thing is that when you work on an array, use the expressions in the same loop not in a different one.

Doggo recoloring:

* Here we checked if any of the character is repeated or not. If repeated then it can be recolored.
* If we take the strlen then for for loop It makes 10^5x10^5 which is tough in case of c.
* So we did is, if the numbered character is less than 27 then we did math. If that was more 27 we still checked from 0 to 27, as it is said that characters are the lower case a to z. sums up sort of. This idea was by ashraf. I was doing my way and got tle.

Juicer:

* Tried a lot with array but there was wa in 5th test.
* Then helped the fact that, we will take a size and sum and if sum is less than d we don’t make it zero and keep summing until it’s greater than d. then counter.
* We also made an expression but that showed wa too.

Carrot cakes:

* Just create an expression that’s all….
* I had to go a lot. Though I could have guessed the proper expression somehow.

Anton and letters:

* Counter change and normal math expressions.

Free ice cream:

* Here when took the loop it indicates the lines.
* In those lines there was a character array and that always placed in the zeroth position. So that was supposed to be noticed.

Team Olympiad:

* New thing. Increasing index with different variables.
* Check again.

Berland Fair:

* Here firstly we checked if the values of a[i] is smaller than a particular defined number.
* Here we didn’t generate an I,j loop check, because that might create complexity as a[i] is huge in amount almost 2\*10^5. It becomes O(2) then.
* Then used some calculation.
* Couldn’t use direct increment function….as there’s a for loop…complexity arises up to a high level. Like more than the I,j loop check.