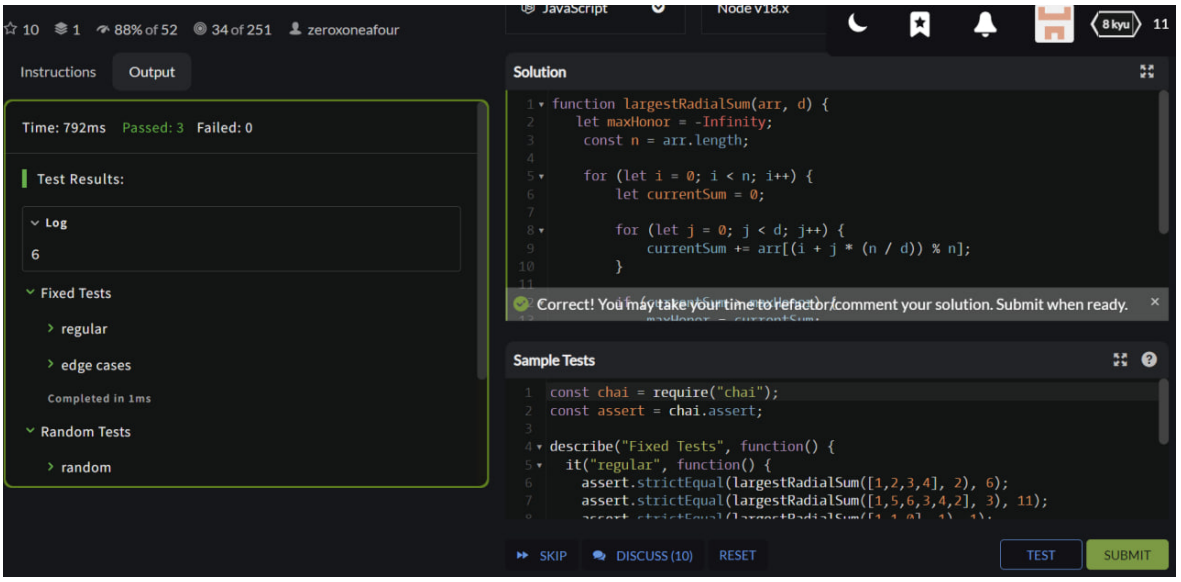
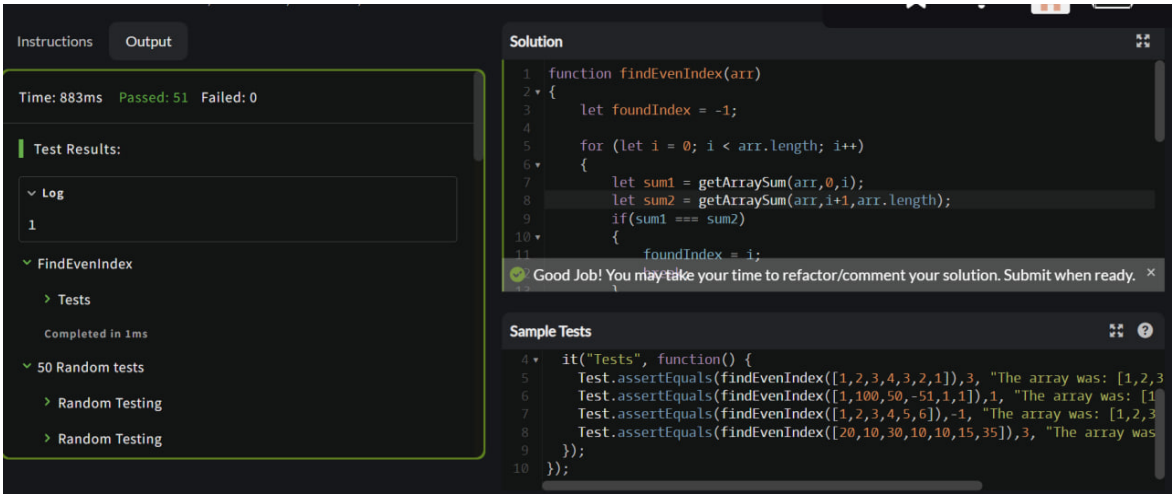
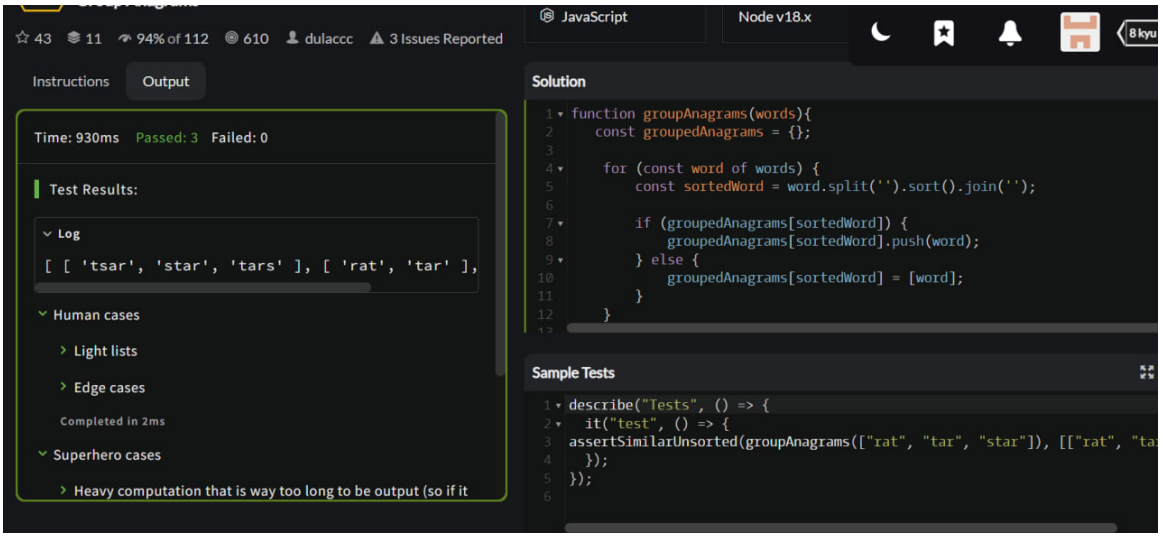
Task1



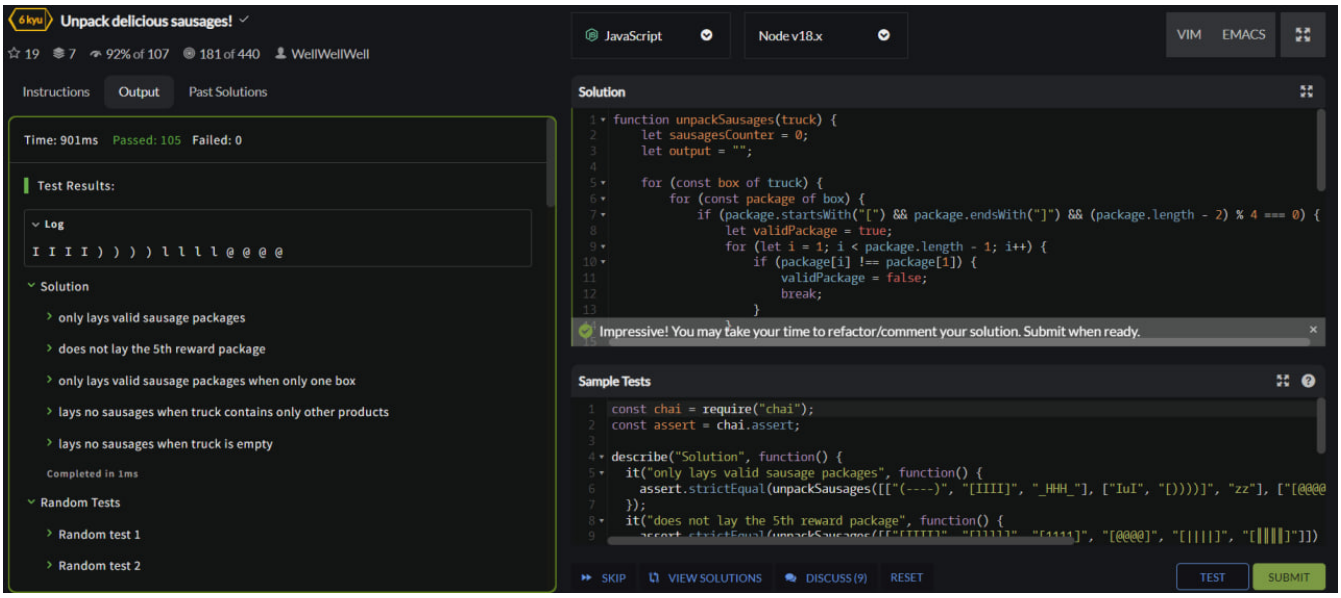
Task2



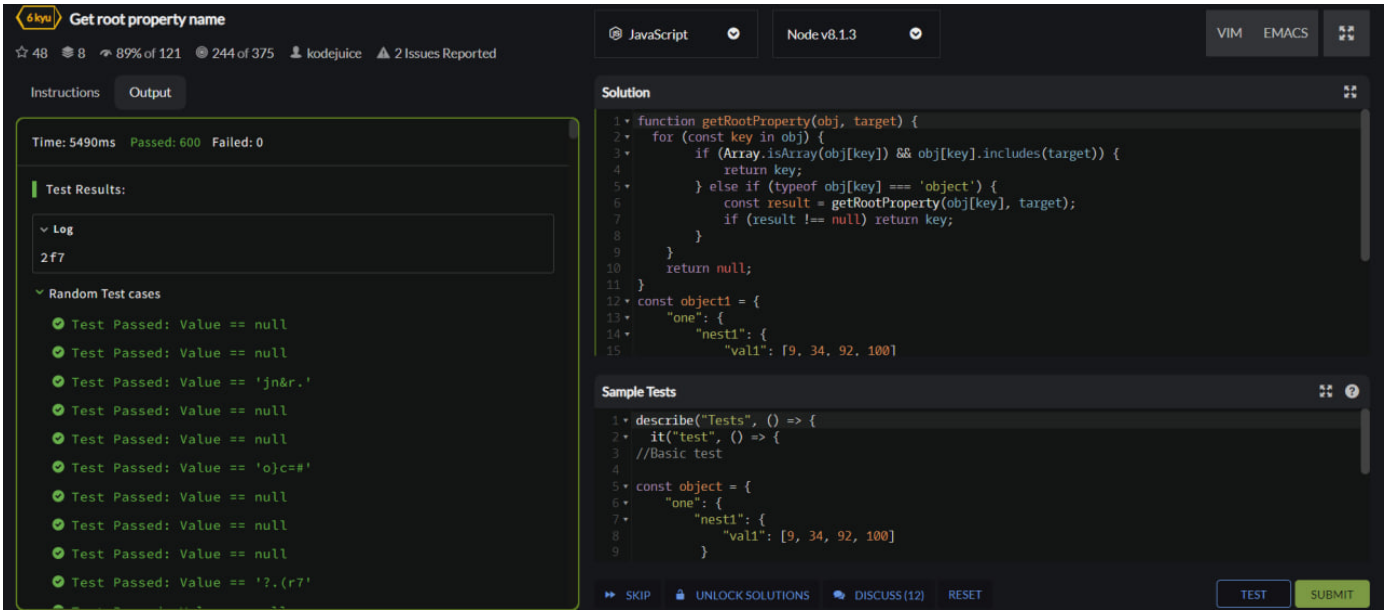
Task3



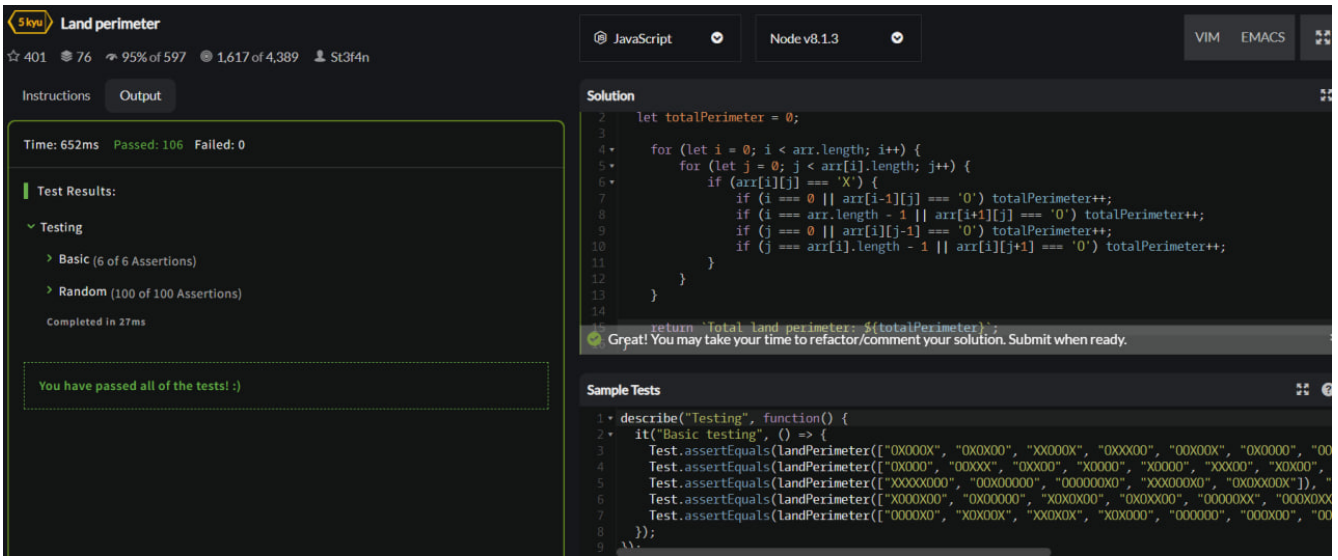
Task 4


6 kyu
Unpack delicious sausages!
19
7
92% of 107
181 of 440
WellWellWell
Instructions
Output
Past Solutions
Time: 901ms
Passed: 105
Failed: 0
Test Results:
Log
I I I I)))) l l l l @ @ @ @
Solution
only lays valid sausage packages
does not lay the 5th reward package
only lays valid sausage packages when only one box
lays no sausages when truck contains only other products
lays no sausages when truck is empty
Completed in 1ms
Random Tests
Random test 1
Random test 2
JavaScript
Node v18.x
VIM
EMACS
Solution
function unpackSausages(truck) {
let sausagesCounter = 0;
let output = "";
for (const box of truck) {
for (const package of box) {
if (package.startsWith("[") && package.endsWith("]") && (package.length - 2) % 4 === 0) {
let validPackage = true;
for (let i = 1; i < package.length - 1; i++) {
if (package[i] !== package[1]) {
validPackage = false;
break;
}
Impressive! You may take your time to refactor/comment your solution. Submit when ready.
Sample Tests
const chai = require("chai");
const assert = chai.assert;
describe("Solution", function() {
it("only lays valid sausage packages", function() {
assert.strictEqual(unpackSausages([["(----)", "[IIII]", "_HHH_"], ["IuI", "[))))]", "zz"], ["[@@@@
});
it("does not lay the 5th reward package", function() {
SKIP
VIEW SOLUTIONS
DISCUSS (9)
RESET
TEST
SUBMIT

Task5


6 kyu
Get root property name
48
8
89% of 121
244 of 375
kodejuice
2 Issues Reported
Instructions
Output
Time: 5490ms
Passed: 600
Failed: 0
Test Results:
Log
2f7
Random Test cases
Test Passed: Value == null
Test Passed: Value == null
Test Passed: Value == 'jn&r.'
Test Passed: Value == null
Test Passed: Value == null
Test Passed: Value == 'o}c=#'
Test Passed: Value == null
Test Passed: Value == null
Test Passed: Value == null
Test Passed: Value == '?.(r7'
JavaScript
Node v8.1.3
VIM
EMACS
Solution
function getRootProperty(obj, target) {
for (const key in obj) {
if (Array.isArray(obj[key]) && obj[key].includes(target)) {
return key;
} else if (typeof obj[key] === 'object') {
const result = getRootProperty(obj[key], target);
if (result !== null) return key;
}
}
return null;
}
const object1 = {
"one": {
"nest1": {
"val1": [9, 34, 92, 100]
Sample Tests
describe("Tests", () => {
it("test", () => {
//Basic test
const object = {
"one": {
"nest1": {
"val1": [9, 34, 92, 100]
}
SKIP
UNLOCK SOLUTIONS
DISCUSS (12)
RESET
TEST
SUBMIT

Task6


5 kyu
Land perimeter
401 76 95% of 597 1,617 of 4,389 St3f4n
Instructions
Output
Time: 652ms Passed: 106 Failed: 0
Test Results:
Testing
Basic (6 of 6 Assertions)
Random (100 of 100 Assertions)
Completed in 27ms
You have passed all of the tests! :)
JavaScript
Node v8.1.3
VIM
EMACS
Solution
let totalPerimeter = 0;
for (let i = 0; i < arr.length; i++) {
for (let j = 0; j < arr[i].length; j++) {
if (arr[i][j] === 'X') {
if (i === 0 || arr[i-1][j] === '0') totalPerimeter++;
if (i === arr.length - 1 || arr[i+1][j] === '0') totalPerimeter++;
if (j === 0 || arr[i][j-1] === '0') totalPerimeter++;
if (j === arr[i].length - 1 || arr[i][j+1] === '0') totalPerimeter++;
}
}
}
return `Total land perimeter: ${totalPerimeter}`;
Great! You may take your time to refactor/comment your solution. Submit when ready.
Sample Tests
describe("Testing", function() {
it("Basic testing", () => {
Test.assertEquals(landPerimeter(["0X000X", "0X0X00", "XX000X", "0XXX00", "00X00X", "0X0000",
Test.assertEquals(landPerimeter(["0X000", "00XXX", "0XX00", "X0000", "X0000", "XXX00", "X0X00",
Test.assertEquals(landPerimeter(["XXXXX000", "00X00000", "000000X0", "XXX000X0", "0X0XX00X"]),
Test.assertEquals(landPerimeter(["X000X00", "0X00000", "X0X0X00", "0X0XX00", "00000XX", "000X0X
Test.assertEquals(landPerimeter(["0000X0", "X0X00X", "XX0X0X", "X0X000", "000000", "000X00",
});