Разумов Дмитрий Александрович

Ссылка на github с решениями: <https://github.com/DiMoNiO04/testirovanie>

Резюме: https://hh.ru/resume/351132abff0ba41a120039ed1f443968746761

## ***ЗАДАНИЯ НА СООБРАЗИТЕЛЬНОСТЬ***

#### 1 задание

Есть 90 человек. Каждый из этих 90 человек относится к одной из двух категорий. Одни на все вопросы говорят правду, другие — всегда врут. У каждого ровно 1 любимое блюдо из списка (макароны/пельмени/вареники). Каждому задают три вопроса:

* *Ваше любимое блюдо — макароны?*
* *Ваше любимое блюдо — пельмени?*
* *Ваше любимое блюдо — вареники?*

Результаты получились такие:

* на первый вопрос утвердительно ответило 45 человек,
* на второй — 35,
* на третий — 30.

Сколько человек всегда говорят правду?

Решение 1-ого задания:

Х - человек всегда говорят правду

У - человек всегда говорят неправду

90 - всего человек =>

Результаты 35(пельмени) + 30(вареники) + 45(макароны) = 110

Так как все отвечают либо правду, либо неправду на все вопросы, и из 3 вопросов каждому только один верно, а два неверно, то

70 человек всегда говорит правду, и 20 человек всегда говорит неправду

Ответ: 70 человек говорит всегда правду

#### 2 задание

Есть числа от трех до одиннадцати. Есть квадрат — 3 на 3 клетки:

а б в

г д е

ё ж з

*(Буквы ничего не значат, кроме того, что числа не повторяются)*

Можно ли расставить числа в клетки квадрата таким образом, что перемножения чисел в строках дает тот же результат, что и произведение чисел в столбцах с теми же номерами?

Если можно — расставьте, если нельзя — объясните почему.

Решение 2-ого задания:

Имеются числа: 3, 4, 5, 6, 7, 8, 9, 10, 11

Найдем пары кратных друг другу чисел: 3 и 6, 4 и 8, 5 и 10. Должны располагаться в разных углах напротив друг друга.

Числа, которые остались: 7, 9, 11 - Должны составлять главную диагональ матрицы

7 3 8 = 168

6 9 5 = 270

4 10 11 = 440

= = =

1 2 4

6 7 4

7 0 0

Ответ: 7 3 8

6 9 5

4 10 11

#### 3 задание

Ученые разработали новый материал неизвестной прочности. Они знают, что материал разбивается при падении с высоты от 1 метра до 5 000 метров. Но не знают, с какой именно высоты. Чтобы определить прочность, ученые поднимают предмет на некоторую высоту и сбрасывают его оттуда. Их задача — определить, начиная с какой именно высоты предмет начнет разбиваться.

Специальная платформа, с помощью которой они осуществляют эксперимент, скидывает предмет только с дискретных высот (1, 2, 3 ... 4999, 5000 метров — платформа не может скинуть предмет, например, с 2,5 метров. Точности в 1 метр ученым вполне достаточно). При падении с высоты "n" метров предмет уничтожается. Если же его сбрасывали с высоты ниже "n", то его можно использовать в повторных экспериментах.

Нужно АБСОЛЮТНО ТОЧНО найти ту высоту, начиная с которой предметы разрушаются. Сделать это нужно за МИНИМАЛЬНО возможное число экспериментов. У ученых при этом всего 2 предмета, но они абсолютно одинаковые. Каким образом этого можно достигнуть? Сколько экспериментов при этом максимально потребуется?

Решение 3-ого задания:

Промежуток ответа [1, 5000] метров

N метров и выше - уничтожили

Ниже N метров- еще можно использовать в дальнейших экспериментах.

2 одинаковых предмета.

Изначально предположил, что можно использовать бинарный поиск и максимально количество шагов 11-12, но если мы сбросим первый раз с 2500, а потом с 1250, а искомое расстояние оказалось меньше, то мы не сможем выяснить, так как израсходуем весь материал.

Поэтому возьмем шаг за 100 метров и будем постепенно увеличивать высоту с 1 метра до 5000. Если высота, к примеру, 501 метров предмет не разбился (так как по условию если меньше искомой высоты можно использовать повторно), а на высоте 601 метров разбился, то у нас остался один предмет и искомая высота с 501 до 601 метров, и далее мы идем с шагом 1 метр начиная с 501, до момента пока второй предмет не разобьется. Максимально количество экспериментов при таком методе будет при высоте 4999 метров, идя с шагом 100 до 5000 метров используем 50 попыток и найдем высоту от 4900 до 5000, и затем идя с шагом 1 до 4999 задействуем еще 99 попыток. Итого максимально может потребоваться 149 попыток.

Ответ: ***Каким образом этого можно достигнуть?***  Шаг 100 метров пока не разобьём первый предмет, а затем шаг 1 метр пока не найдем нужную высоту. ***Сколько экспериментов при этом максимально потребуется?*** 149 попыток

#### 4 задание

В университетскую столовую пришли трое студентов купить пирожки. Один хочет купить треть всех имеющихся в столовой пирожков и еще 2 пирожка. Другой хочет купить четверть от всех имеющихся пирожков и ещё 3 пирожка. А третий хочет купить пятую часть всех пирожков и ещё 8 пирожков. Если все трое купят столько пирожков, сколько планируют, то в столовой как раз закончатся все пирожки. Сколько пирожков есть в столовой?

Решение 4-ого задания:

X - всего пирожков

) - первый студент; () - второй студент; () - третий студент

Отсюда найдем х

- Количество всех пирожков

22 пирожка купил первый студент, второй купил 18, третий купил 20

Ответ: 60 пирожков

***ЗАДАЧИ ПО ПРОГРАММИРОВАНИЮ***

#### Задача 1

Написать метод/функцию, который/которая на вход принимает массив городов. В качестве результата возвращает строку, где города разделены запятыми, а в конце стоит точка.

Решение 1-ой задачи:

//Выполнение данной задачи заняло 12минут (из которых половина ушла на рефакторинг и написание проверок)

arrCitiesBelarus = ['Гродно', 'Гомель', 'Могилев', 'Минск', 'Солигорск'];

arrCitiesRussia = ['Москва', 'Петербург', 'Сочи', 'Краснодар'];

arrCitiesGermany = ['Берлин', 'Мюнхен', 'Франкфурт'];

noCities = 123; arrEmpty = [];

let result = '';

const MSG\_ARR\_EMPTY = 'Пустой массив';

const NO\_ARR = 'Не массив. На вход должен принимается массив';

function outputCitiesInString(arr){

if(Array.isArray(arr)){

(arr.length === 0) ? result = MSG\_ARR\_EMPTY : result = arr.join(', ') + '.';

}else {

result = NO\_ARR;

}

return result;

}

console.log(outputCitiesInString(arrCitiesBelarus));

console.log(outputCitiesInString(arrCitiesRussia));

console.log(outputCitiesInString(arrCitiesGermany));

console.log(outputCitiesInString(noCities));

console.log(outputCitiesInString(arrEmpty));

#### Задача 2

Написать метод/функцию, который/которая на вход принимает число (float), а на выходе получает число, округленное до пятерок.

Решение 2-ой задачи:

//Выполнение данной задачи заняло 7 минут + рефакторинг

const roundToFive = (number) => result = Math.round(number / 5) \* 5;

console.log(roundToFive(27))

console.log(roundToFive(27.8))

console.log(roundToFive(41.7))

Задача 3

Написать метод/функцию, который/которая на вход принимает число (int), а на выходе выдает слово “компьютер” в падеже, соответствующем указанному количеству. Например, «25 компьютеров», «41 компьютер», «1048 компьютеров».

Решение 3-ой задачи:

//Выполнение задания заняло 22 минуты + рефакторинг

const ERR\_MSG = 'Не является числом или число отрицательное или число не является целым'

let result = '';

function wordInCase(number){

if(Number(number) && number >= 0 && Number.isInteger(number)){

if((number === 1) || (number % 10 === 1 && number > 20)) result = number + ' компьютер'

if((number >= 5 && number <= 20) || (number%10 >= 5) || (number%10 === 0))result = number + ' компьютеров'

if((number < 5 && number % 10 < 5) || (number > 20 && number % 10 < 5)) result = number + ' компьютера';

}

else result = ERR\_MSG;

return result;

}

console.log(wordInCase(22))

console.log(wordInCase(41))

console.log(wordInCase(1))

console.log(wordInCase(1048))

console.log(wordInCase(1048.9))

#### Задача 4

Написать метод/функцию, который/которая на вход принимает целое число, а на выходе возвращает то, является ли число простым (не имеет делителей кроме 1 и самого себя).

Решение 4-ой задачи:

//Выполнение задания заняло 14 минуты + рефакторинг

let countDivisor = 2; //Так как число изначально делиться уже на 1 и на само себя

let result = '';

const ERR\_MSG = 'Число не является целым или не является числом';

function isPrimeNumber(number){

if(Number.isInteger(number)){

for(let divisor = Math.round(number / 2); divisor >= 2; divisor--){

if(number % divisor === 0) countDivisor++;

}

(countDivisor !== 2) ? result = `Число ${number} не является простым числом` : result = `Число ${number} является простым числом`

}else{

result = ERR\_MSG;

}

return result;

}

console.log(isPrimeNumber(7))

#### Задача 5

Написать метод, который определяет, какие элементы присутствуют в двух экземплярах в каждом из массивов (= в двух и более, причем в каждом). На вход подаются два массива. На выходе массив с необходимыми совпадениями.

Решение 5-ой задачи:

//Выполнение задания заняло больше часа, из-за того, что не мог понять, как сделать метод в методе, нашел только такой ход решения

class Repeat{

constructor(arrFirst, arrSecond){

this.arrFirst = arrFirst;

this.arrSecond = arrSecond;

}

getRepeatElementsFromTwoArrs(arrFirst, arrSecond){

function getRepeatElementForOneArr(arrFirst){

let result = [];

for(let i = 0; i < arrFirst.length; i++){

for(let j = i+ 1; j < arrFirst.length; j++){

if(arrFirst[i] === arrFirst[j] && !result.includes(arrFirst[i])){

result.push(arrFirst[i])

}

}

}

return result;

}

function getRepeatElements(arrFirst, arrSecond){

let result = []

let newArrFirst = getRepeatElementForOneArr(arrFirst);

let newArrSecond = getRepeatElementForOneArr(arrSecond);

for(let i=0; i< newArrFirst.length; i++){

for(let j=0; j<newArrSecond.length; j++){

if(newArrFirst[i] === newArrSecond[j]){

result.push(newArrFirst[i])

}

}

}

return result.sort();

}

return getRepeatElements(arrFirst, arrSecond)

}

}

let firstArr = new Repeat([7, 17, 1, 9, 1, 17, 56, 56, 23], [56, 17, 17, 1, 23, 34, 23, 1, 8, 1]);

console.log(firstArr.getRepeatElementsFromTwoArrs([7, 17, 1, 9, 1, 17, 56, 56, 23], [56, 17, 17, 1, 23, 34, 23, 1, 8, 1]))

let secondArr = new Repeat([7, 17, 1, 9, 1, 56, 56, 23], [56, 17, 17, 1, 23, 34, 23, 1, 8, 1]);

console.log(secondArr.getRepeatElementsFromTwoArrs([7, 17, 1, 9, 1, 56, 56, 23], [56, 17, 17, 1, 23, 34, 23, 1, 8, 1]))

let thirdArr = new Repeat([7, 9, 1, 56, 56, 23], [56, 17, 17, 23, 34, 23, 1, 8, 1]);

console.log(thirdArr.getRepeatElementsFromTwoArrs([7, 9, 1, 56, 56, 23], [56, 17, 17, 23, 34, 23, 1, 8, 1]))

#### Задача 6

Написать метод, который в консоль выводит таблицу умножения. На вход метод получает число, до которого выводит таблицу умножения. В консоли должна появиться таблица. Например, если на вход пришло число 5, то получим

Решение 6-ой задачи:

//Выполнение данной задачи заняло 35 минут (из-за метода вывода по ТЗ) + рефакторинг

class MultiplicationTable{

constructor(number){

this.number = number;

}

getMutliplicationTable(number){

let result = '';

for(let multiplierFirst = 0; multiplierFirst <= number; multiplierFirst++){

(multiplierFirst === 0) ? result += ' ' : result += multiplierFirst + ' ';

}

for(let multiplierFirst = 1; multiplierFirst <= number ; multiplierFirst++){

result += '\n' + multiplierFirst + ' ';

for(let multiplierSecond = 1; multiplierSecond <= number; multiplierSecond++){

if((String(multiplierFirst \* multiplierSecond).length && String(multiplierFirst \* (multiplierSecond + 1)).length) === 1){

result += (multiplierFirst \* multiplierSecond) + ' ';

}else{

result += (multiplierFirst \* multiplierSecond) + ' ';

}

}

}

return result

}

}

let number = new MultiplicationTable();

console.log(number.getMutliplicationTable(5))

Разумов Дмитрий Александрович

Ссылка на github с решениями: <https://github.com/DiMoNiO04/testirovanie>

Резюме: https://hh.ru/resume/351132abff0ba41a120039ed1f443968746761