

Задания Javascript

Задания должны исполняться в браузере Google Chrome последней версии.

Результат выполнения задания должен быть выведен в консоль браузера.

1. Существует некий замок с n вращающихся дисков. На каждом диске написаны цифры с 0 до 9. Чтобы открыть замок нужно поворачивать диски до тех пор, пока они не образуют секретную комбинацию. Существует только одна секретная комбинация. За одно действие можно прокрутить только один диск и только на одну цифру вперед или назад. Требуется посчитать минимальное кол-во действий, для того, чтобы открыть замок.

Входные данные:

- Задано n дисков ($1 \leq n \leq 10$)
- Задано текущее положение дисков
- Задана секретная комбинация

Выходные данные:

- Вывести минимальное кол-во действий, чтобы открыть замок

Пример:

Входные данные:

- Задано 6 дисков
- Текущее положение дисков 821906
- Секретная комбинация 797083

Выходные данные:

- 14

Предоставьте исходный код решения.

```
const usePrompt = () => {
  const error = "Неверные данные! Попробуйте еще раз.\n"
  const regCount = new RegExp(/^([1-9]|10)$/);
  const regDisks = new RegExp(/^\d+$/);

  let disksCount = prompt("Количество дисков от 1 до 10: ");
  while (!regCount.test(disksCount)) {
    disksCount = prompt(`${error}Количество дисков от 1 до 10: `);
  }

  let disks = prompt("Текущее положение дисков: ");
  while (!regDisks.test(disks) || disks.length !== +disksCount) {
    disks = prompt(`${error}Текущее положение дисков: `);
  }

  let disksSecret = prompt("Секретная комбинация: ");
```

```

while (!regDisks.test(disksSecret) || disksSecret.length !== +disksCount)
{
    disksSecret = prompt(`${error}Секретная комбинация: `)
}

return { disksSecret, disks: +disks, disksSecret: +disksSecret }
}

const { disks, disksSecret } = usePrompt()

// const disks = 821906
// const disksSecret = 797083

const steps = Array.from(String(disks)).reduce((steps, disk, i) => {
    const difference = Math.abs(+disk - +String(disksSecret)[i])

    return (steps += Math.min(difference, 10 - difference))
}, 0)

console.log("Минимальное количество действий: ", steps)

```

- Имеется n групп рабочих. Каждая группа состоит из s ($1 \leq s_i \leq 4$) рабочих. Необходимо перевезти на строительный объект все группы рабочих на машинах. Каждая машина может перевезти только 4х рабочих, при этом группы делить нельзя, каждую группы нужно перевезти целиком. Требуется посчитать минимальное количество машин необходимых для перевозки рабочих.

Входные:

- Задано n групп рабочих ($1 \leq n \leq 100$)
- Задана строка, разделенная одиночным пробелом, в которой содержится последовательность целых чисел s_1, s_2, \dots, s_n ($1 \leq s_i \leq 4$). s_i — количество рабочих в i -ой группе.

Выходные данные:

- Минимальное необходимое количество машин необходимых для перевозки.

Пример:

Входные:

- 8 групп
- 2 3 4 4 1 2 3 1

Выходные данные:

- 5

Входные:

- 12 групп
- 4 1 4 2 1 2 3 1 3 1 3 1

Выходные данные:

- 7

Предоставьте исходный код решения.

```
const usePrompt = () => {
  const error = "Неверные данные! Попробуйте еще раз.\n"
  const reg = new RegExp(/([1-4] )?[1-4]/)

  let groupCount = prompt("Количество групп от 1 до 100: ")
  while (+groupCount < 1 || +groupCount > 100 || isNaN(+groupCount)) {
    groupCount = prompt(`${error}Количество групп от 1 до 100: `)
  }

  let groups = prompt("Группы по порядку через пробел: ")
  while (!reg.test(groups) || groups.split(" ").length !== +groupCount) {
    groups = prompt(`${error}Группы по порядку через пробел: `)
  }

  return { groupCount, groups }
}

const { groups } = usePrompt()

// const groupCount = 8
// const groups = "2 3 4 4 1 2 3 1"

let groupsList = groups.split(" ")

let single = 0
let couples = 0
let triplets = 0
let quadruplets = 0

groupsList.forEach((g) => {
  if (g === "1") single += 1
  if (g === "2") couples += 1
  if (g === "3") triplets += 1
  if (g === "4") quadruplets += 1
})

const oneWithTriplets = Math.min(single, triplets)

// если остаются тройни они едут одни в машине
triplets -= oneWithTriplets

// если остаются одинокие они стакаются с парами
single -= oneWithTriplets
couples += Math.floor(single / 2)
```

```
// если осталась нераспределенная двойня или нераспределенный одинокий
// то нужна отдельная машина (если двойня и один - тоже поместятся)
single = couples % 2 || single % 2 ? 1 : 0

const cars =
  quadruplets + oneWithTriplets + triplets + Math.floor(couples / 2) +
  single

console.log("Минимальное количество машин: ", cars)
```