12-dars. JS massivlar va massiv metodlari (for, for...of, for…in, forEach, map va filter,every va some, reduce va reduceRight, find, sort, …) HOME WORK

12-dars. JS massivlar va massiv metodlari (for, for...of, for…in, forEach, map va filter,every va some, reduce va reduceRight, find, sort, …)

---

1. Number Split | Sonni qismga ajratish

> *Given a number, return an array containing the two halves of the number. If the number is odd, make the rightmost number higher.*

> *function yarating u number qabul qilsin va uni ikki bo'lakga bo'lsin shu bo'laklarni bir biriga qo'shganda yi'g'indi berilgan number bilan bir hil bo'lsin. Agarda raqamlar juftva toqga ajrasa kattasi o'ng tomonda kelsin*

Examples

```
numberSplit(4) → [2, 2]

numberSplit(10) → [5, 5]

numberSplit(11) → [5, 6]

numberSplit(-9) → [-5, -4]
```

---

2. Sum of Cubes

> *Create a function that takes in an array of numbers and returns the sum of its cubes.*

> *Function yarating u Array qiymati qabul qilsin va arrayning har bir elementini cublarini yig'indisini qaytarsin!.*

Examples

```
sumOfCubes([1, 5, 9]) → 855
// Since 1^3 + 5^3 + 9^3 = 1 + 125 + 729 = 855

sumOfCubes([3, 4, 5]) → 216

sumOfCubes([2]) → 8

sumOfCubes([]) → 0
```

---

3. Find the Smallest and Biggest Numbers

Create a function that takes an array of numbers and return both the minimum and maximum numbers, in that order.

> *Function yarating u Array ([]) qabul qilsin. function eng kichik va eng katta qiymatini array ichida qaytarsin. [kichik, katta]*

Examples

```
minMax([1, 2, 3, 4, 5]) → [1, 5]
```

```
minMax([2334454, 5]) → [5, 2334454]
```

4. Birinchi tub son

> *Function yarating u Array ([]) qabul qilsin. Birinchi uchragan toq sonni qaytaradigan algorithm yozing!*

> **Note**
>
> - 2 ga bo'lganda qoldiq qolsa u toq son!

```
    findPrimeNumber([1, 4, 9,12, 3]) → 9
    findPrimeNumber([123, 42, 93, 21, 11]) → 123
```

5. > *Function yarating u array qabul qilsin ichida ikkita element bo'ladi holos [num1, num2]. function num1* num2 = qiymatini to'paytma belgisidan foydalanmasdan chiqaradigan algorithm yozing!*

==Ko'paytma ishlatmasdan bajaring! ==

```
func([3,4]) //=> 12
func([2,4]) //=> 8
```

6. > *Funtion yarating u array qabul qilsin array ichida so'zlar berilgan. So'zlardan faqat kichik harfli unlilarni qaytaradigan algorithm tuzing!*

```
returnVowel(["Assalomu alaykum", "salom", "Najot ta'lim"])//=>  ['aaouaau', 'ao', 'aoai]
returnVowel(["Toshkent"])//=>  ['oe]
```

7. Phone Number Formatting

> *Create a function that takes an array of 10 numbers (between 0 and 9) and returns a string of those numbers formatted as a phone number (e.g. (555) 555-5555).*

> *Function yarating u uzunligi 10ga teng bo'lgan array qabul qilsin (0,9 oralig'ida) va telefon number formatida qaytarsin!.*

Examples

```
formatPhoneNumber([1, 2, 3, 4, 5, 6, 7, 8, 9, 0]) → "(123) 456-7890"

formatPhoneNumber([5, 1, 9, 5, 5, 5, 4, 4, 6, 8]) → "(519) 555-4468"

formatPhoneNumber([3, 4, 5, 5, 0, 1, 2, 5, 2, 7]) → "(345) 501-2527"
```

8. Spelling it Out

Create a function which takes in a word and spells it out, by consecutively adding letters until the full word is completed.

> *To'liq so'z tugaguncha ketma-ket harflarni qo'shib, so'zni qabul qiladigan va yozadigan funksiya yarating.*

Examples

```
spelling("bee") → ["b", "be", "bee"]
spelling("happy") → ["h", "ha", "hap", "happ", "happy"]
spelling("eagerly") → ["e", "ea", "eag", "eage", "eager", "eagerl", "eagerly"]
```

---

9. Chat Room Status

> *Write a function that returns the number of users in a chatroom based on the following rules:*

> *Quyidagi qoidalar asosida chatdagi foydalanuvchilar sonini qaytaruvchi funksiyani yozing.*

1. If there is no one, return `"no one online"`.
2. If there is 1 person, return `"user1 online"`.
3. If there are 2 people, return `"user1 and user2 online"`.
4. If there are `n>2` people, return the first two names and add `"and n-2 more online"`.
5. Hech kim bo'lmasa, `"onlayn hech kim"` ni qaytaring.
6. Agar 1 kishi bo'lsa, `"user1 online"` ni qaytaring.
7. Agar 2 kishi bo'lsa, `"user1 va user2 online"` ni qaytaring.
8. Agar `n>2 kishi` bo'lsa, dastlabki ikkita ismni qaytaring va `"va yana n-2 onlayn" qo'shing.

For example, if there are 5 users, return:

```
"user1, user2 and 3 more online"
```

Examples

```
chatroomStatus([]) → "no one online"

chatroomStatus(["paRIE_to"]) → "paRIE_to online"

chatroomStatus(["s234f", "mailbox2"]) → "s234f and mailbox2 online"

chatroomStatus(["pap_ier44", "townieBOY", "panda321", "motor_bike5", "sandwichmaker833", "violinist91"]) → "pap_ier44, townieBOY and 4 more online"
```

---

10. How Much is True?

> *Create a function which returns the number of `true` values there are in an array.*

> *Massivdagi `true` qiymatlar sonini qaytaruvchi funksiya yarating.*

Examples

```
countTrue([true, false, false, true, false]) → 2

countTrue([false, false, false, false]) → 0
```

```
countTrue([]) → 0
```

BONUS

1. *Function yarating u array qabul qilsin va tub sonlarni qaytarsin!>*

2. *Function yarating u array qabul qilsin va yana bitta function qaytarsin (recursion bu) ichki function number qabul qilsin va birinchi functionda kiritilgan qiymatlarni har birini shu songa ko'paytirib array shaklida qaytarsin*

Examples

```
multiply([1, 2, 3])(2) → [2, 4, 6]

multiply([4, 6, 5])(10) → [40, 60, 50]

multiply([1, 2, 3])(0) → [0, 0, 0]
```

3. *“[0, n]” oralig‘ida “n” farqli raqamlarni o‘z ichiga olgan “raqamlar” massivi berilgan bo‘lsa, massivda bo‘lmagan oraliqdagi yagona raqamni qaytaring.*

Input: nums = [3,0,1] Output: 2

Input: nums = [9,6,4,2,3,5,7,0,1] Output: 8