

1. \$\$\$ Input ile daxil olunan 2 ededin cemi.
2. \$\$\$ Input ile daxil olunan musbet ədədlərin cemi.
3. Input ile daxil olunan ədədin sıfır, müsbət yoxsa mənfi olduğunu tapın.
4. Input ile daxil olunan 2 eded arasında olan ən böyük ededi tapın
5. \$\$\$ Input ile daxil olunan ədədin 5-e tam bölünüb bölünmədiyini tapın.
6. \$\$\$ Input ile daxil olunan ədədin cüt yoxsa tək olduğunu tapın.
7. Daxil olunan ededin özündən rəqəmlərinin cəmini çıxırıq
8. Diskriminantın tapılması
9. Daxil olunan ededin rəqəmlərindən necəsinin cüt olduğunu tapın
10. Eddi sadə vuruqlara ayırmaq
11. Eddən özünün rəqəmləri cəmini çıxmaq tək ədəd alınana kimi
12. 10 ilə 200 arasında ədədlərin 3 qatının 2 eded artığının 5-e bölünən ədədləri göstərin
13. 1-100 arasındakı cüt ədədlərin toplamını tapın
14. 1-100 arasındakı cüt ədədlərin toplamının mükəmməl ədəd olub-olmadığını tapın.
15. \$\$\$ Ekranə 5 dəfə Adını yazdır
16. Input ilə daxil edilən adı 5 dəfə ekranə yazdır
17. \$\$\$ 1-dən 20-yə qədər ədədləri ekranə yazdır
18. Input ilə daxil edilən adı, Input ilə daxil edilən ədəd dəfə ekranə yazdır
19. Input ilə daxil olunan n sayda ədəddin cemi.
20. 1-dən Input ilə daxil edilən ədədə qədər cüt ədədləri tapın
21. Input ilə daxil edilən 5 əddən mənfi olanların toplamını, cüt olanların vuruğunu tapın,
22. \$\$\$ Factorial-in hesablanması
23. ***** 5 fəndən ümumi ortalama çıxsin. Dərslər input ilə daxil edilsin və 0dan 100-e qədər qiymət verilsin. Eddi

ortalama tapılsın 5 derse göre. Eger ortalama 90+ ise A, 81-90 ise B, 71-80 ise C, 61-70 ise D, 51-60 ise E, 50den asagi ise Kesilmissiniz ekrana cixsin. Istifadeci menfi eded daxil ede bilmez

24. Input ile 5 eded daxil edin. 7'ye beraber olanların sayini tapib ekrana yazdırin
25. Input ile daxil olunan ədədin sadə bolenlerini cixartsin
26. Daxil etdiyiniz ededde nece 0-var
27. *** 3 rəqəmli eded daxil edirik ededin evveline ve sonuna string ile 3 elave edirik
28. Input ile daxil olunan ədədin bölənlərini tapın.
29. **Input ile daxil olunan ədədin bölənlərinin sayını tapın.**
30. Input ile daxil olunan ədədin bölənlərinin cəmini tapın.
31. Input ile daxil olunan ədədin bölənlərinin hasilini tapın.
32. **Input ile daxil olunan ədədin neçə rəqəmli olduğunu tapın.**
33. **Input ile daxil olunan ədədin rəqəmlərinin cəmini tapın**
34. Input ile daxil olunan ədədin rəqəmlərinin hasilini tapın.
35. \$\$\$ **Input ile daxil olunan ədədin rəqəmlərindən ən böyüyünü tapın.**
36. \$\$\$ Input ile daxil olunan ədədin sadə yoxsa mürəkkəb olduğunu tapın.
37. Input ile daxil olunan ədədin rəqəmlərindən tam ədədləri göstərin.
38. **Input ile daxil olunan ədədin rəqəmlərindən cüt ədədləri göstərin.**
39. Input ile daxil olunan ədədin rəqəmlərindən sadə ədədləri göstərin.
40. Input ile daxil olunan ədədin rəqəmlərindən mükəmməl ədəd olub olmadığını tapın.
41. *** Input ile daxil olunan ədədin rəqəmlərin sayi ve string ile tərsini tapın.

42. Input ile daxil olunan ədədin tərsinin bölənlərini göstərin.
43. Input ile daxil olunan 1 ci ədədi daxil olunan 2 ci ədədin qüvvətinə yüksəldin.
44. Input ile daxil olunan ədədin rəqəmlərinin kvadratlari cəmini tapın.
45. *** Input ile daxil olunan ədədin 3 ededi string ile yan-yanə yigilsin. Misal ucun: 3, 45, 10
46. Input ile daxil olunan ədədin 3 ededi string ile yan-yanə ardıcilliq ile yigilsin. Misal ucun: 3, 10, 45
47. İstifadəçidən alınan n sayda ədədi array'i toplayın və sıralayın
48. Daxil edilen ededlerin ededi ortasini tapın
49. İstifadəçidən alınan n sayda ededden necesinin tam eded olduğunu tapın
50. İstifadəçidən alınan n sayda ededden necesinin sade eded olduğunu tapın
51. İstifadəçidən alınan n sayda ededden necesinin murekkeb eded olduğunu tapın
52. Daxil olunan 2 eded arasında olan en boyuk sade ededi tapın
53. a_b ->10_40
54. \$\$\$ n - > 9 -> 0 1 1 2 3 5 8 13 21 (*Fibonacci*)
55. Input ile daxil olunan ededin reqlerinde ilk sade ededi tapın
56. Array olsun, target olsun, target-den sonraki array-in index'ini tapın (**home work**)
57. Daxil olunan ededden sonraki ilk n sade ededi tapın
58. İstifadəçinin daxil etdiyi ededin kok altini tap
59. Array icerisinde en kicik ededi tapın (**home work**)
60. Array icersisinde sonu 9 olan nece elemet var. (**home work**)
61. Array icerisinde nece cut element var
62. Array icerisinde nece tek element var
63. Arraylerin hamisin cut olub olmadigina baxın.

64. Array icerisindeki tek ededlerin cemini tapin.
65. Array icerisindeki tam ededlerin cemini tapin (while ile). (home work)
66. Array icerisinde nece element oldugunu tapin.
67. Istifadeci 0 daxil edenedek edelerin cemini tapin.
68. ***** Vurma cedveli (string methodu ile)
69. Arraydaki ededlerin ededi ortasini tapin.
70. Input ile daxil olunan 3 ededden ortancili tapan funksiya yazin.
71. Array icine soz a herfi ile baslayani tapin. (home work)
72. Arrayda m herfi ile biten sozleri cixardin.
73. Array icerisinde 0-9 kimi reqemler var, hansisa 1 reqem yoxdur onu tapin
74. Arrayda sozde butun herfleri boyuk olub olmadigina baxin
75. Arrayda guzgu ededleri tapsin
76. a = 4, b = 5

```
* * * *  
* * * *  
* * * *  
* * * *  
* * * *
```

```
* * * *  
*      *  
*      *  
*      *  
* * * *
```

77. n = 7

```
1  
1 2  
1 2 3  
1 2 3 4
```

- 1 2 3 4 5
- 1 2 3 4 5 6
- 1 2 3 4 5 6 7
78. n sayda random herf password
79. 1 - 10 kimi ededleri cixart (while)
80. Istifadeci 0 daxil edenedek ededleri arrayda goster (while)
81. Eded arraya cevir
82. Input ile daxil olunan ededin ceminden evvelki ededin cixilmesi (while)
83. Textdeki soz de herf nece defe tekrarlanib.
84. Arraydaki tekrarlanan reqemlerden birincini cixartsin.
85. Arrayda text icindeki sozlerin ilk herfini boyuk cixartsin
86. Arrayda stringlerin ilk stringi boyuke cevir
87. Arrayda stringlerin son herfini boyuke cevirin
88. Arrayde ededleri tersine cevirmek
89. Arrayin icindeki ededlerin cemi cutdurse --true, tekdirse-- false return etsin
90. Arrayda her iki ededden bir iki eded arasinda reqemlerin cemi
91. Qarışıq array verilib, yalnızca ədədləri çap etdirin.
92. Qarışıq array verilib, yalnızca stringləri çap etdirin.
93. Arrayda metodlar (en azi 3 method ile edilsin) ile ededleri 5 vahid artirmaq
94. Input ile daxil olunan stringler icerisindeki saitlerin sayini tapin
95. Arrayde 6 olanlari "-" ile evez edin
96. Array icerisindeki ən böyük ən kiçik ədədin tapılması

97. N - integerə $(2n + 1)$ görə paxlava generator

N = 5 olquda asagidaki kimi vizual yaranmalidir.

// *

```

//      ***
//      *****
//      ****
//      ****
//      ****
//      ****
//      ****
//      ***
//      *

```

98. 2 eded olsun (a, b). 3cu ededi istifade etmeden ededlerin yerini deyisin.

99. 2 array olsun 1 dovr, 1 sert ile arraylerdeki diff elementleri tapin

```

Var arr1 = [1, 2, 3, 4]
Var arr2 = [1, 2, 3, 4, 5]
Var diffArr = [5]

```

1. Find the first repeating character in a string.

```

function firstRepeatingChar(str) {
    const charSet = new Set();
    for (const char of str) {
        if (charSet.has(char)) {
            return char;
        }
        charSet.add(char);
    }
    return null;
}
console.log(firstRepeatingChar("abcdab")); // 'a'

```

2. Flatten a nested array.

```

function flattenArray(arr) {

```

```

    return arr.reduce((acc, val) => Array.isArray(val) ?
acc.concat(flattenArray(val)) : acc.concat(val), []);
}
console.log(flattenArray([1, [2, 3, [4, 5]], 6])); //
[1,2,3,4,5,6]

```

3. Capitalize the first letter of each word in a string.

```

function capitalizeWords(str) {
    return str.split(' ').map(word =>
word.charAt(0).toUpperCase() + word.slice(1)).join(' ');
}
console.log(capitalizeWords("hello world")); // "Hello World"

```

4. Find the missing number in an array.

```

function findMissingNumber(arr, n) {
    const total = n * (n + 1) / 2;
    const sum = arr.reduce((a, b) => a + b);
    return total - sum;
}
console.log(findMissingNumber([1, 2, 4, 6, 5, 3, 8], 8)); // 7

```

5. Check if a word is an anagram of another.

```

function isAnagram(str1, str2) {
    return str1.split('').sort().join('') ===
str2.split('').sort().join('');
}
console.log(isAnagram("listen", "silent")); // true

```

6. Sum all numbers in an object (nested or not).

```

function sumObj(obj) {
    let sum = 0;

```

```

    for (const key in obj) {
        if (typeof obj[key] === "number") {
            sum += obj[key];
        } else if (typeof obj[key] === "object") {
            sum += sumObj(obj[key]);
        }
    }
    return sum;
}
console.log(sumObj({ a: 1, b: { c: 2, d: { e: 3 } } })); // 6

```

7. Return chunked array.

```

function chunkArray(arr, size) {
    const chunked = [];
    for (let i = 0; i < arr.length; i += size) {
        chunked.push(arr.slice(i, i + size));
    }
    return chunked;
}
console.log(chunkArray([1, 2, 3, 4, 5], 2)); //
[[1,2],[3,4],[5]]

```

15. Return the longest word in a sentence.

```

console.log(longestWord("The quick brown fox jumped over the
lazy dog")); // 'jumped'

```

16. Factorial of a number.

```

console.log(factorial(5)); // 120

```

17. Convert a string to snake case.

```

console.log(toSnakeCase("Hello World!")); // 'hello_world'

```


18. Return an array's unique values.

```
console.log(uniqueValues([1,2,2,3,4,4])); // [1,2,3,4]
```

19. Find the sum of all even numbers up to a given number.

```
console.log(sumEvenNumbers(10)); // 30
```

20. Check if all items in an array are identical.

```
console.log(allSame([1,1,1,1])); // true
```

21. Return the middle character(s) of a string.

```
console.log(middleChar("middle")); // 'dd'
```

22. Transpose a matrix.

```
console.log(transposeMatrix([[1,2,3], [4,5,6], [7,8,9]])); //  
[[1,4,7], [2,5,8], [3,6,9]]
```

23. Find the greatest common divisor of two numbers.

```
console.log(gcd(14, 28)); // 14
```

24. Remove falsey values from an array.

```
console.log(removeFalsey([0, 1, false, 2, '', 3])); // [1,2,3]
```

25. Find the shortest word in a string.

```
console.log(shortestWord("The quick brown fox jumped over the  
lazy dog")); // 'The'
```

26. Generate an array with a given range and step.

```
console.log(rangeWithStep(1, 10, 2)); // [1,3,5,7,9]
```

27. Return the most frequent number in an array.

```
console.log(mostFrequentNumber([1, 2, 3, 3, 3, 4, 5, 5])); // 3
```

28. Check if a string is a valid number.

```
console.log(isValidNumber("123")); // true
```

29. Find the intersection of two arrays.

```
console.log(intersectArrays([1,2,2,3,4], [2,3,5,6])); // [2,3]
```

30. Check if a string contains all letters of the alphabet.

```
console.log(hasAllLetters("The quick brown fox jumps over a lazy dog")); // true
```

31. Deep clone an object.

```
const obj1 = { a: { b: 1 } };  
const obj2 = deepClone(obj1);  
console.log(obj1.a.b === obj2.a.b); // false
```

32. Merge two sorted arrays.

```
console.log(mergeSortedArrays([1,3,5], [2,4,6])); //  
[1,2,3,4,5,6]
```

34. Check if two strings are a rotation of each other.

```
console.log(areRotations("abc", "cab")); // true
```

36. Return the power of a number without using Math.pow().

```
1. function removeDuplicates(array) {  
  const unique = {};  
  const result = [];  
  for (const element of array) {  
    if (!unique[element]) {  
      result.push(element);  
      unique[element] = true;  
    }  
  }  
  return result;  
}
```

```
const newArray = removeDuplicates(array);  
console.log(newArray); // [1, 2, 3, 4]
```

```
2. function reverse(num) {  
  
  let reversedNum = ""  
  let stringNum = num.toString()  
  
  for (let i = stringNum.length - 1; i >= 0; i--) {  
    reversedNum += stringNum[i]  
  }  
  
  return +reversedNum  
  
}
```

```
console.log(reverse(1234))
```

```
3. function maxCharacter(str) {  
  const charMap = {};
```

```
let maxChar = '';
let maxCount = 0;

// Count frequency of each character
for (const char of str) {
  if (charMap[char]) {
    charMap[char]++;
  } else {
    charMap[char] = 1;
  }
}

// Find the character with the highest frequency
for (const char in charMap) {
  if (charMap[char] > maxCount) {
    maxChar = char;
    maxCount = charMap[char];
  }
}

return maxChar;
}

console.log(maxCharacter("Hello World!"))
```

4. let num = 78

```
let second = (num % 10) // 8
let first = num - second // 78 - 8 = 70
```

```
var obj = {
  0: "sifir",
  1: "bir",
  2: "iki",
  3: "uc",
```

```

4: "dort",
5: "bes",
6: "alti",
7: "yeddi",
8: "sekkiz",
9: "doqquz",
10: "on",
20: "iyirmi",
30: "otuz",
40: "qirx",
50: "elli",
60: "altmis",
70: "yetmis",
80: "seksen",
90: "doxsan"
}

for (let i = 0; i < 90; i++) {
  if (obj[i]) {
    if (i == first) {
      var _first = obj[i]
    }
    if (i == second) {
      var _second = obj[i]
    }
  }
}

console.log(_first + " " + _second)

function flattenArray(arr) {
  return arr.reduce((acc, val) => Array.isArray(val) ?
    acc.concat(flattenArray(val)) : acc.concat(val), []);
}

```

```
console.log(flattenArray([1, [2, 3, [4, 5]], 6])); //
[1,2,3,4,5,6]
```

```
function capitalizeWords(str) {
    return str.split(' ').map(word =>
word.charAt(0).toUpperCase() + word.slice(1)).join(' ');
}
console.log(capitalizeWords("hello world")); // "Hello World"
```

```
function isAnagram(str1, str2) {
    return str1.split('').sort().join('') ===
str2.split('').sort().join('');
}
console.log(isAnagram("listen", "silent")); // true
```

```
function sumObj(obj) {
    let sum = 0;
    for (const key in obj) {
        if (typeof obj[key] === "number") {
            sum += obj[key];
        } else if (typeof obj[key] === "object") {
            sum += sumObj(obj[key]);
        }
    }
    return sum;
}
console.log(sumObj({ a: 1, b: { c: 2, d: { e: 3 } } })); // 6
```

```
function longestWord(sen) {
    return sen.split(' ').sort((a, b) => b.length -
a.length)[0];
}
console.log(longestWord("The quick brown fox jumped over the
lazy dog")); // 'jumped'
```

```
function mergeSortedArrays(arr1, arr2) {  
    return arr1.concat(arr2).sort((a,b) => a-b);  
}  
console.log(mergeSortedArrays([1,3,5], [2,4,6])); //  
[1,2,3,4,5,6]
```

```
function longestWord(str) {  
    const words = str.split(' ');  
    return words.reduce((a, b) => a.length > b.length ? a : b);  
}
```

```
console.log(longestWord("The quick brown fox")); // "quick"
```

```
function mergeObjects(obj1, obj2) {  
    return { ...obj1, ...obj2 };  
}
```

```
console.log(mergeObjects({a: 1, b: 2}, {c: 3, d: 4})); // {a: 1,  
b: 2, c: 3, d: 4}
```

```
function countOccurrences(arr) {  
    return arr.reduce((acc, curr) => {  
        acc[curr] = (acc[curr] || 0) + 1;  
        return acc;  
    }, {});  
}
```

```
console.log(countOccurrences(["a", "b", "a", "c", "b"])); // {a:  
2, b: 2, c: 1}
```

```
function sortByLength(arr) {  
    return arr.sort((a, b) => a.length - b.length);  
}
```

```
console.log(sortByLength(["apple", "kiwi", "banana"])); //  
["kiwi", "apple", "banana"]
```

```
function getKeysByValue(obj, condition) {  
  return Object.keys(obj).filter(key => condition(obj[key]));  
}
```

```
console.log(getKeysByValue({a: 10, b: 20, c: 30}, val => val >  
15)); // ["b", "c"]
```

```
function removeValues(arr, ...values) {  
  return arr.filter(item => !values.includes(item));  
}
```

```
console.log(removeValues([1, 2, 3, 4, 5], 2, 4)); // [1, 3, 5]
```

```
function groupByParity(arr) {  
  return arr.reduce((acc, num) => {  
    acc[num % 2 === 0 ? 'even' : 'odd'].push(num);  
    return acc;  
  }, {even: [], odd: []});  
}
```

```
console.log(groupByParity([1, 2, 3, 4])); // {even: [2, 4], odd:  
[1, 3]}
```

22 en cox verilen suallar (22 common questions in interviews)

- <https://medium.com/siliconwat/algorithms-in-javascript-b0bed68f4038>

<https://leetcode.com/problemset/all/?difficulty=EASY&page=1&listId=wpwgkgt>

<https://replit.com/@AmilAbdullazade/JSTest#index.js>

Final (Algo & JS)

- 1) Prompdan daxil olan ədədin faktorialini tapın
- 2) Arrayin içərisindəkiləri kiçikdən böyüyə göstərin
- 3) Arraydaki sadə ədədləri göstərin
- 4) Input ile daxil olan stringdə olan saitleri ve onların sayını tapın
- 5) Promptdan n sayda ədəd daxil olacaq daxil olunan ədədlər arasında ən böyüyü tapın
- 6) Verilən cümlədə bütün sözlərin baş hərfini böyüdün
- 7) Verilən ədəddən sonrakı ilk sadə ədədi tapın
- 8) Verilən array sonu 9 bitən ədəd varmı? Varsa True, yoxsa False ekrana çıxardın
- 9) Daxil etdiyimiz array-de neçə cüt element var. Bunu tapan funksiya yazın
- 10) Array içərisində 0-9 kimi rəqəmlər var, 1 çatmır onu tapın
- 11) Daxil olunan stringin ilk hərfini böyüyə çevir
- 12) Array-da for-la ədədləri 5 vahid artırmaq
- 13) $n = 7$
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
- 14) $a = 4, b = 5$
* * * *
* *
* *
* *
* * * *
- 15) Fibonacci sequence-ni recursion ile edin
- 16) ES6 ve ondan sonrakılar nədir?

- 17) Closure funksiyası haqqında ətraflı məlumat yazın. (misal verin)
- 18) Hoisting dedikdə nə başa düşürsüz?
- 19) HOF nəyə deyilir?
- 20) Callback nədir?
- 21) Data types neçə dənədir və hansılardır?
- 22) Let ve const fərqi nələrdir?
- 23) Recursion funksiya nədir və if else olmadan yazmaq mümkündür?
- 24) setTimeout() asinxron yoxsa sinxron funksiyaadır ? Ve misal ile izah edin
- 25) Hansı method arrayı string'e çevirir?
- 26) forEach ve map arasindaki ferq ?
- 27) $O(\log n)$ nece teying olunur ?
- 28) "Hello my friend. How are you ?" bu cumlede en cox istifade olunan her ve ya herifleri ekranda gosterin
- 29) Memoized Fibonacci. Yeni fibonacci ucun funksiya yardin. Sadece men eyni funcsiyani yeniden cagirarsa eyni parametrlere onun ucun hesablama aparmamlidir. Cunki evvelce edib. Hint.
 1. cache= {}
 2. !cache[target] or cache[target]