

// Input: [{"Bob","87"}, {"Mike", "35"}, {"Bob", "52"}, {"Jason","35"}, {"Mike", "55"}, {"Jessica", "99"}]
// Output: 99Explanation: Since Jessica's average is greater than Bob's, Mike's and Jason's average.

```
const findMaxAverage = (students) => {  
  const stdData = {};  
  let maxAvg = -1;  
  
  for(const [name, marks] of students) {  
    const mark = parseInt(marks, 10);  
    if(!stdData[name]) {  
      stdData[name] = {totalMarks: 0, count: 0}  
    }  
  
    stdData[name].totalMarks += mark  
    stdData[name].count += 1  
  }  
  
  console.log(stdData);  
  
  for(const name in stdData) {  
    const {totalMarks, count} = stdData[name]  
    const avg = totalMarks / count;  
  
    if(avg > maxAvg) {  
      maxAvg = avg  
    }  
  }  
  
  return maxAvg  
}  
  
const Input = [  
  ["Bob","98"],  
  ["Mike", "35"],  
  ["Bob", "96"],  
  ["Jason","35"],  
  ["Mike", "55"],  
  ["Jessica", "91"],  
  ["Bob", "0"]  
]  
  
console.log(findMaxAverage(Input));
```

```

/*
[1,2,3,4,5,6] , sum is 6

[2,4],[1,5]

*/

function arraySum(arr, target) {
  arr.sort((a,b) => a-b);
  const pairs = [];
  let left = 0;
  let right = arr.length-1;

  while(left < right) {
    const sum = arr[left] + arr[right];
    // console.log("start", left, right, arr[left], arr[right], sum, pairs);

    if(sum === target) {
      pairs.push([arr[left], arr[right]])
      left++
      right--
    }
    else if( sum < target) {
      left++;
    }else {
      right--
    }
    // console.log("end", left, right, arr[left], arr[right], sum, pairs);
  }
  return pairs;
}

// console.log(arraySum([1,2,3,4,5,6,7], 7))
// console.log(arraySum([1,2,3,4,5,6], 6))
// console.log(arraySum([1, 2, 5, 8, 9, 10, 13, 14], 15))

```

```

function arraySumWithSet(arr, target) {
  const arrSet = new Set();
  const pairs = [];
  for(const num of arr) {
    const diff = target - num;

    if(arrSet.has(diff)) {
      pairs.push([num, diff])
    }

    arrSet.add(num)
  }
  return pairs;
}

console.log(arraySumWithSet([1, 2, 5, 8, 9, 10, 13, 14], 15))

```



```

// Given an integer array nums, return true
// if there exists a triple of indices (i, j, k) such that i < j < k and nums[i] < nums[j] < nums[k]. If no such indices exists, return false.

// Input: nums = [1,2,3,4,5]; Output: true

function checkIndices(nums) {
  if (nums.length < 3) return false;
  for (let i = 0; i < nums.length - 2; i++) {
    for (let j = i + 1; j < nums.length - 1; j++) {
      for (let k = j; k < nums.length; k++) {
        if (nums[i] < nums[j] && nums[j] < nums[k]) {
          console.log("true");
          return true;
        }
      }
    }
  }
  console.log("false");
  return false;
}

function checkIndices2(nums) {
  const n = nums.length;
  if (n < 3) return false;

  const leftMin = new Array(n);
  leftMin[0] = nums[0];

  // Precomputing the left minimum of the array
  for (let i = 1; i < n; i++) {
    leftMin[i] = Math.min(leftMin[i - 1], nums[i]);
  }

  const rightMax = new Array(n);
  rightMax[n - 1] = nums[n - 1];

  // Precomputing the right max of the array
  for (let i = n - 2; i >= 0; i--) {
    rightMax[i] = Math.max(rightMax[i + 1], nums[i]);
  }

  for (let i = 1; i < n - 1; i++) {
    if (leftMin[i - 1] < nums[i] && nums[i] < rightMax[i + 1]) {
      console.log("true");
      return true;
    }
  }

  console.log("false");
  return false;
}

// checkIndices2([1,2,3,4,5])
// checkIndices2([5,4,3,2,1])
// checkIndices2([5,1,2,3,4])
// checkIndices2([1,2,5,4,3])

```

```
/*
```

Input: n = 5, arr[] = {"3", "30", "34", "5", "9"}

Output: "9534330"

Explanation: Given numbers are {"3", "30", "34", "5", "9"}, the arrangement "9534330" gives the largest value.

```
*/
```

```
const largestNumber = (arr) => {  
  return arr.sort((a, b) => b + a - (a + b)).join("");  
};
```

```
// console.log(largestNumber(["3", "30", "34", "5", "9"]));
```

```
// Given an integer n, return true if it is a power of three. Otherwise, return false.
```

```
// An integer n is a power of three, if there exists an integer x such that n == 3x.
```

```
const isPowerOf3 = (n) => {  
  if (n < 1) return false;  
  while (n % 3 === 0) {  
    n = n / 3;  
  }  
  return n === 1;  
};
```

```
console.log(isPowerOf3(1024));
```