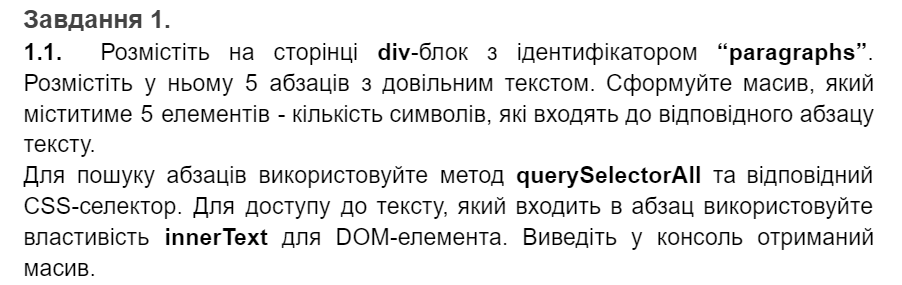
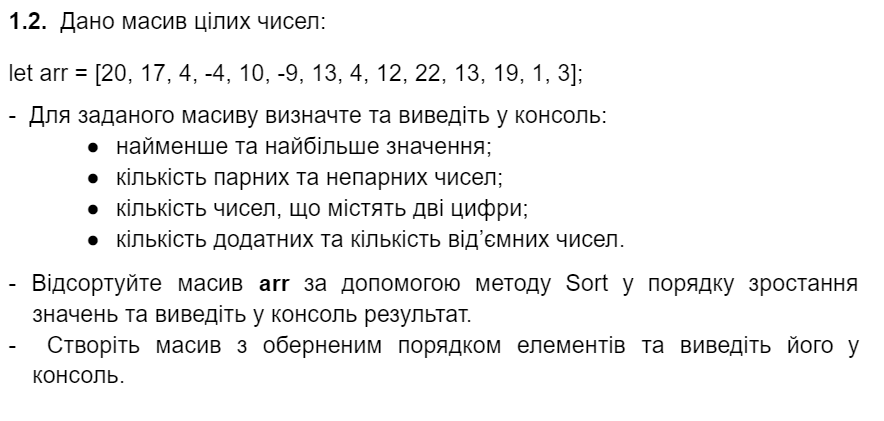
**Лабораторна робота №3**

**Мета роботи:**

**Хід роботи:**

***Завдання 1***.

Лістинг програми:

*const* divId = document.getElementById('paragraphs');

*const* paragraphs = divId.querySelectorAll('p');

*const* counts = [];

paragraphs.forEach((paragraph) *=>* {

*const* text = paragraph.innerText;

*const* char = text.length;

    counts.push(char);

});

console.log("Завдання 1.2.")

*//1.2.*

*let* arr = [20, 17, 4, -4, 10, -9, 13, 4, 12, 22, 13, 19, 1, 3];

*let* min = Math.min(...arr);

console.log("Min: " + min);

*let* max = Math.max(...arr);

console.log("Max: " + max);

*let* parni = 0;

*let* neparni = 0;

*for* (*let* number of arr)  {

*if*(number % 2 === 0){

      parni++;

   }

*else*{

      neparni++;

   }

}

console.log("К-ть парних чисел:" + parni);

console.log("К-ть непарних чисел:" + neparni);

*let* pos = 0;

*let* neg = 0;

*for* (*let* number of arr) {

*if* (number > 0) {

    pos++;

  } *else* *if* (number < 0) {

    neg++;

  }

}

console.log("К-ть додатних чисел:", pos);

console.log("К-ть від’ємних чисел:", neg);

*let* twoNum = 0;

*for* (*let* number of arr) {

*if* (Math.abs(number) >= 10 && Math.abs(number) <= 99) {

   twoNum++;

  }

}

console.log("К-ть чисел, що містять дві цифри:", twoNum);

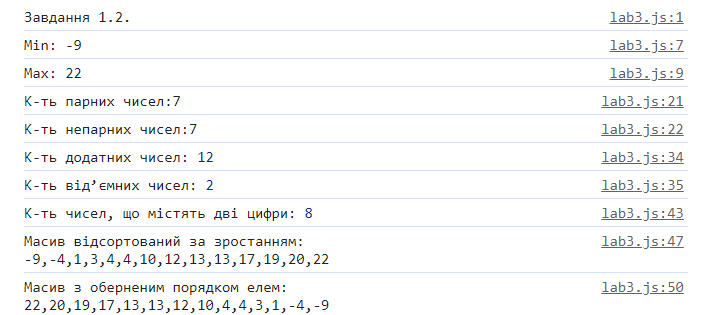
arr.sort((a,b) *=>* a-b)

console.log("Масив відсортований за зростанням: " + arr)

*let* reversArr = arr.slice().reverse().join();

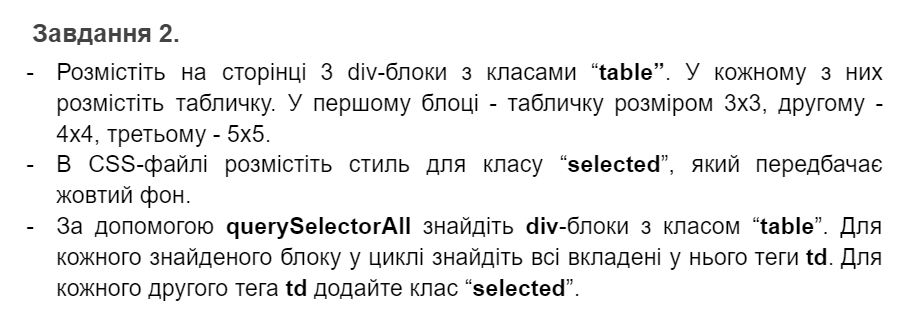
console.log("Масив з оберненим порядком елем:", reversArr);

Результати виконання:





***Завдання 2***.



Лістинг програми:

window.onload = *function*() {

*const* divTable = document.querySelectorAll('.table');

divTable.forEach((divTable) *=>* {

*const* tds = divTable.querySelectorAll('td');

    tds.forEach((td, index) *=>* {

*if* (index % 2 !== 0) {

            td.classList.add('selected');

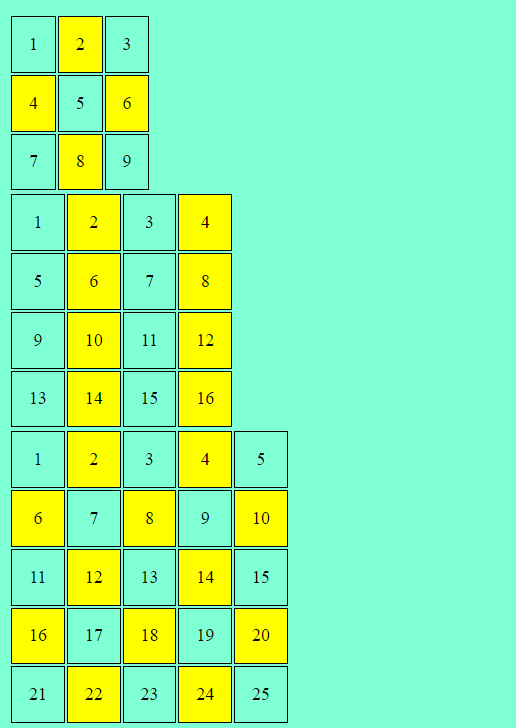
        }

    });

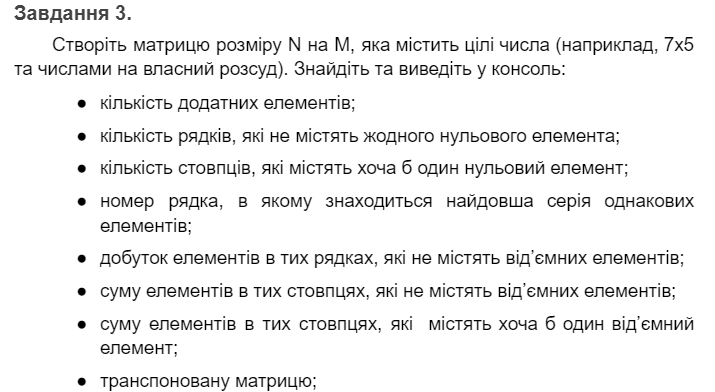
});

}

Результати виконання:



***Завдання 3***.



Лістинг програми:

console.log("Матриця" );

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

*let* matrixTask = [

   [-2, 1, -6, 8, 7],

   [-1, -4, 3, 2, 0],

   [0, 9, 4, -2, -6],

   [5, 9, 1, -2, 0],

   [0, 1, 3, 3, 4],

   [6, 0, 7, 4, 9],

   [8, -7, 2, -4, 3]

];

*function* printmatrixTask(matrixTask) {

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

     console.log(matrixTask[i]);

   }

 };

printmatrixTask(matrixTask);

*let* posNum = 0;

*let* noZeromatrixTaskRow = 0;*// рядки без нуля*

*let*  zeroCol = 0;*//стовп з нулями*

*let* product = 1;

*let* sumPos = 0;

*let* sunmNeg = 0;

*let* transmatrixTask = [];

*let* longmatrixTaskRow = 0;

*let* longDouble = 0;

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

*let* matrixTaskRow = matrixTask[i];

*let* double = 0;

*let* hZero = false;

*let* hNeg = false;

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

*let* box = matrixTaskRow[j];

*if* (box > 0) {

           posNum++;

       }

*if* (box === 0) {

           hZero = true;

*if* (matrixTask.map(matrixTaskRow *=>* matrixTaskRow[j]).some(box *=>* box === 0)) {

              zeroCol++;

           }

       }

*if* (box < 0) {

           hNeg = true;

       }

*if* (j > 0 && box === matrixTaskRow[j - 1]) {

           double++;

       } *else* {

           double = 1;

       }

*if* (double > longDouble) {

           longDouble = double;

           longmatrixTaskRow = i;

       }

*if* (!transmatrixTask[j]) {

           transmatrixTask[j] = [];

       }

       transmatrixTask[j][i] = box;

   }

*if* (!hZero) {

       noZeromatrixTaskRow++;

   }

*if* (!hNeg) {

       product \*= matrixTaskRow.reduce((a, b) *=>* a \* b, 1);

   }

*if* (hNeg) {

       sunmNeg += matrixTaskRow.reduce((a, b) *=>* a + b, 0);

   } *else* {

       sumPos += matrixTaskRow.reduce((a, b) *=>* a + b, 0);

   }

}

console.log('К-ть додатних елем:', posNum);

console.log('К-ть рядків, які не містять жодного нульового елемента:', noZeromatrixTaskRow);

console.log('К-ть стовпців, які містять хоча б один нульовий елемент:',  zeroCol);

console.log('Номер рядка, в якому знаходиться найдовша серія однакових елем:', longmatrixTaskRow);

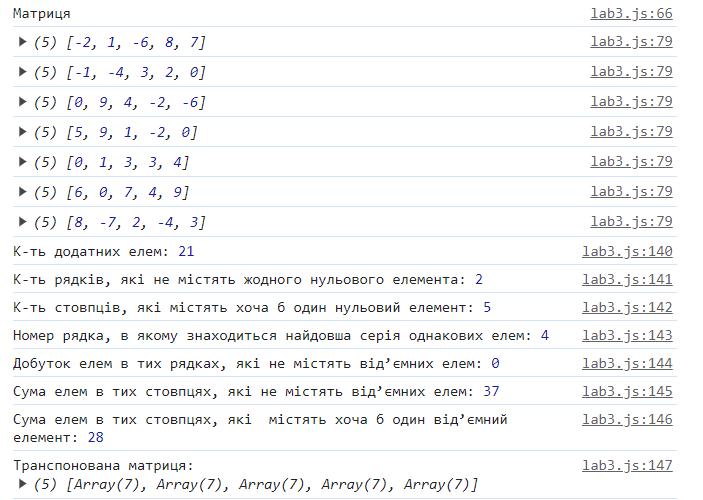
console.log('Добуток елем в тих рядках, які не містять від’ємних елем:', product);

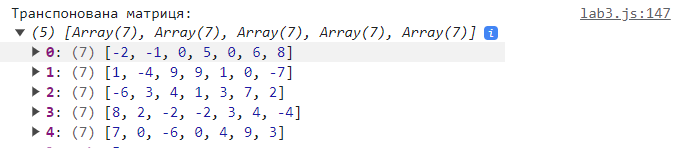
console.log('Сума елем в тих стовпцях, які не містять від’ємних елем:', sumPos);

console.log('Сума елем в тих стовпцях, які  містять хоча б один від’ємний елемент:', sunmNeg);

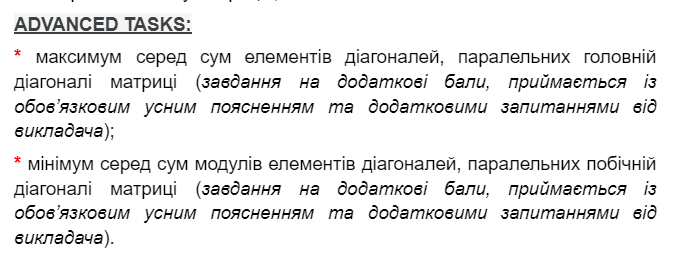
console.log('Транспонована матриця:', transmatrixTask);

Результати виконання:





**Додаткове завдання:**

****

*Лістинг коду:*

*function* mainDia(matrixTask) {

*let* maxSumMainDia = 0;

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

*let* sum = 0;

*for* (*let* j = 0; i + j < matrixTask.length; j++) {

       sum += matrixTask[i + j][j];

     }

*if* (sum > maxSumMainDia) {

       maxSumMainDia = sum;

     }

   }

*return* maxSumMainDia;

 }

*function* SecondDia(matrixTask) {

*let* minSumSecondDia = 0;

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

*let* sum = 0;

*for* (*let* j = 0; j <= i; j++) {

       sum += Math.abs(matrixTask[i - j][j]);

     }

*if* (sum < minSumSecondDia) {

       minSumSecondDia = sum;

     }

   }

*return* minSumSecondDia;

 }

*const* maxSumMainDia = mainDia(matrixTask);

*const* minSumSecondDia = SecondDia(matrixTask);

 console.log("Максимум ", maxSumMainDia);

 console.log("Мінімум ", minSumSecondDia);

*Результат:*

**