# 1. What is the expected running time of the following C# code? Explain why. Assume the array's size is n.

long Compute(int[] arr)

{

long count = 0;

for (int i=0; i<arr.Length; i++)

{

int start = 0, end = arr.Length-1;

while (start < end)

if (arr[start] < arr[end])

{ start++; count++; }

else

end--;

}

return count;

}

* Runs in quadratic time O(**n2**)
* Влизаме “n” на брой пъти във for цикълa, като всеки път завъртаме while също “n” на брой пъти.

# 2. What is the expected running time of the following C# code? Explain why.

long CalcCount(int[,] matrix)

{

long count = 0;

for (int row=0; row<matrix.GetLength(0); row++)

if (matrix[row, 0] % 2 == 0)

for (int col=0; col<matrix.GetLength(1); col++)

if (matrix[row,col] > 0)

count++;

return count;

}

Assume the input matrix has size of n \* m.

* Runs in quadratic time O(n \* m)
* Влизаме “n – x” на брой пъти във първия for цикъл, след което всеки път влизаме във втория “m”на брой пъти.

# 3.\* What is the expected running time of the following C# code? Explain why.

long CalcSum(int[,] matrix, int row)

{

long sum = 0;

for (int col = 0; col < matrix.GetLength(0); col++)

sum += matrix[row, col];

if (row + 1 < matrix.GetLength(1))

sum += CalcSum(matrix, row + 1);

return sum;

}

Console.WriteLine(CalcSum(matrix, 0));

**Assume the input matrix has size of n \* m.**

* Runs in quadratic time O(**n \* m**)
* Обхождаме колоните “m” брой пъти, след което и редовете “n-row” на брой пъти.