Exercise 3 (JULIA)

Create a 2x4 two dimensional matrix with random floats in it and in the next step determine the biggest element.

First, I generate the Matrix:

In [1]:

```
theMatrix = rand(Float64, (2, 4))
display(theMatrix)

```

```
2×4 Matrix{Float64}:
 0.838737  0.390757  0.0586751  0.0360709
 0.541854  0.566697  0.740464   0.755574
```

Then, I find the position of the biggest element:

In [2]:

```
positionOfMax = argmax(theMatrix)
```

Out[2]:

```
CartesianIndex(1, 1)
```

Finally, I get the element at the position that was found

In [3]:

```
biggestElt = theMatrix[positionOfMax]
```

Out[3]:

```
0.8387369524224816
```

Exercise 4 (JULIA)

1. Create two matrices of the same layout and test if addition and subtraction of the matrix works as expected: C = A + B

First, I create 2 Matrices

In [4]:

```
matrix1 = [12 4 9 74; 65 45 38 3; 76 17 39 5; 65 9 34 65]
println("Matrix 1:")
display(matrix1)
matrix2 = [9 54 23 18; 3 38 45 28; 92 6 37 21; 63 23 94 12]
println("Matrix 2:")
display(matrix2)


```

```
Matrix 1:

4×4 Matrix{Int64}:
 12   4   9  74
 65  45  38   3
 76  17  39   5
 65   9  34  65

Matrix 2:

4×4 Matrix{Int64}:
  9  54  23  18
  3  38  45  28
 92   6  37  21
 63  23  94  12
```

addition

In [5]:

```
additionMatrix = matrix1 + matrix2
println("additionMatrix: ")
display(additionMatrix)
```

```
additionMatrix:

4×4 Matrix{Int64}:
  21  58   32  92
  68  83   83  31
 168  23   76  26
 128  32  128  77
```

Substraction

In [6]:

```
substractionMatrix = matrix1 - matrix2
println("substractionMatrix: ")
display(substractionMatrix)
```

```
substractionMatrix:

4×4 Matrix{Int64}:
   3  -50  -14   56
  62    7   -7  -25
 -16   11    2  -16
   2  -14  -60   53
```

=> The addition and the substraction work as expected => addition or substraction of the elements

2. Now compare matrix multiplication either this way A * B and this way A .* B. Whats the difference?!

In [7]:

```
multiply1 = matrix1*matrix2
multiply2 = matrix1.*matrix2
display(multiply1)
display(multiply2)

```

```
4×4 Matrix{Int64}:
 5610  2556  7745  1405
 4405  5517  5208  3264
 4638  5099  4426  2723
 7835  5551  9268  2916

4×4 Matrix{Int64}:
  108   216   207  1332
  195  1710  1710    84
 6992   102  1443   105
 4095   207  3196   780
```

=> Here we see that the matrices multiplication is not the same as the multiplication of the elements of two matrices.

3. What about matrix division with "/" or ""?!

In [8]:

```
divide1 = matrix1/matrix2
divide2 = matrix1\matrix2
display(divide1)
display(divide2)
```

```
4×4 Matrix{Float64}:
 -1.85676    3.54659   1.34491   -1.67716
  1.43526   -1.22345   0.230804   0.547921
  0.616514  -0.773419  0.488241   0.442118
 -1.27726    2.48532   1.51066   -1.11018

4×4 Matrix{Float64}:
 -17.2328   -13.1552     35.5676   -10.7652
  -8.62887   -3.07468    11.8263    -3.20326
  39.8905    27.165     -73.8882    22.9944
  -1.46899   -0.274653    2.89023   -0.634517
```

=> The two operations are different because: matrix1/matrix2 = matrix1* inv(matrix2) and matrix1\matrix2 = inv(matrix1)*matrix2

In [9]:

```
display(matrix1* inv(matrix2))
display(inv(matrix1)*matrix2)
```

```
4×4 Matrix{Float64}:
 -1.85676    3.54659   1.34491   -1.67716
  1.43526   -1.22345   0.230804   0.547921
  0.616514  -0.773419  0.488241   0.442118
 -1.27726    2.48532   1.51066   -1.11018

4×4 Matrix{Float64}:
 -17.2328   -13.1552     35.5676   -10.7652
  -8.62887   -3.07468    11.8263    -3.20326
  39.8905    27.165     -73.8882    22.9944
  -1.46899   -0.274653    2.89023   -0.634517
```

4. Create a 3x3 integer matrix A with useful numbers. Now try A+1, A-1, A*2, A/2.

In [10]:

```
#3x3 matrix
A = [5 4 9 7; 6 2 8 3; 4 3 7 5]
display(A)
```

```
3×4 Matrix{Int64}:
 5  4  9  7
 6  2  8  3
 4  3  7  5
```

the oparations A+1 and A-1 are not possible and generate errors. To add and integer to or substract an integer from a matrix, one should write A.+1 or A.-1

In [11]:

```
A_plus_1 = A .+ 1
display(A_plus_1)
A_minus_1 = A .- 1
display(A_minus_1)
```

```
3×4 Matrix{Int64}:
 6  5  10  8
 7  3   9  4
 5  4   8  6

3×4 Matrix{Int64}:
 4  3  8  6
 5  1  7  2
 3  2  6  4
```

The multiplication and the division work without any problem.

In [12]:

```
A_mult_2 = A * 2
display(A_mult_2)
A_div_2 = A / 2
display(A_div_2)
```

```
3×4 Matrix{Int64}:
 10  8  18  14
 12  4  16   6
  8  6  14  10

3×4 Matrix{Float64}:
 2.5  2.0  4.5  3.5
 3.0  1.0  4.0  1.5
 2.0  1.5  3.5  2.5
```

5. Now multiply a 3x4 matrix with a suitable (4)vector.

In [13]:

```
matrixM = [5 4 9 7; 6 2 8 3; 4 3 7 5]
vectorV = [2; 4; 7; 3]
multResult = matrixM*vectorV
display(matrixM)
display(vectorV)
display(multResult)
```

```
3×4 Matrix{Int64}:
 5  4  9  7
 6  2  8  3
 4  3  7  5

4-element Vector{Int64}:
 2
 4
 7
 3

3-element Vector{Int64}:
 110
  85
  84
```