Tasks 2024

There is given an array tab with the size MxN (number of rows - number of columns), e.g. int M = 3, N =5;

```
int tab[3][5] = { {1, 100, 8, 90, -1},
                  {0, 1, 2, -2, 100},
                  {10, 11, 12, -21, 100} };
```

Tasks to perform on the exemplary array:

Assume that tab stores int values.

Tasks - display the input array and the output result:

1. find the smallest value in each row.
2. find the smallest value in each column.
3. find the largest value in each column.
4. find the largest value in each row.

Perform similar tasks assuming that tab stores floating-point numbers.

Extra tasks:

1. find the index of the element with the highest / lowest value.
2. find the element whose square value is minimal;
3. find the value of the sum of the squares of the elements for a given array.

Other tasks:

1. Transposition;
2. Square array -> determinant,
3. Reverse array: A*B={1}

1. Transposition - exchange rows with columns; A^T
2. Multiplication of arrays A[MxK] * B[KxN]
3. Multiply A * (A^T) and (A^T) *A;

Examples of the open tasks for the test 1 - summary.

1.

a. Write an algorithm that returns the sum of the decimal digits of a positive integer provided by the user.

b. Write an algorithm that returns the product of the decimal digits of a positive integer provided by the user.

2) In the given array 'tab' of integers/floating-point doubles, find:

a. The smallest/highest value;

b. Index under which the smallest/largest value is located;

3) In the given array of integers/floating-point doubles of size MxN (row x column), search for:

a. The smallest elements in a row;

b. The smallest elements in the column;

c. Indexes of the smallest values in a row/column.

4) For the given array of integer/floating-point double numbers of size MxN (row x column-per), write an algorithm that returns a transposed array (i.e. swaps rows with columns).

5) Write the code of a function/algorithm that reverses the order of elements of a one-dimensional array of real numbers given by the call arguments without using an additional array.

6) Write the code of a function/algorithm that calculates the sum of the squares of the elements on the main diagonal of a two-dimensional square array of real numbers given by the arguments of the call.

7) Write the code of a function/algorithm that returns the number of decimal digits present in the given character array call arguments.

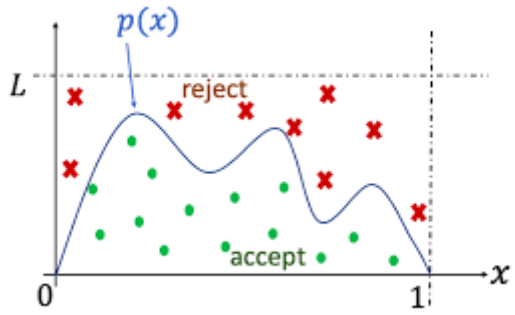
8) Write the code of a function/algorithm that calculates the product of squares of elements on the inverse diagonal of a two-dimensional square array of real numbers given by the arguments of the call.

9) Write the code of a function/algorithm that returns the number of lowercase letters in the character array given as arguments.

Extra tasks

Monte Carlo method – how to calculate the area under function.

$$\int_0^{3.1415} 3 \cdot x^3 \cdot \sin(x)\, dx = \left[-3 \cdot x^3 \cdot \cos(x) + 9 \cdot x^2 \cdot \sin(x) + 18 \cdot x \cdot \cos(x) - 18 \cdot \sin(x) \right]_0^{3.1415} = 36.470161877$$




```
#include <iostream>
#include <cmath>

using namespace std;

double func(double x){
return 3*pow(x,3)*sin(x)
}
```

Calculate area under f(x) between a and b, suppose f(x) > 0 in the range [a,b];

Provide a and ;

Provide N number of iterations;

Check a < b;

If a < b;

Provide number h , h > maximum value of f(x) in the ;

Generate 2 numbers x1 that is a < x1 <b and y that is 0<y1 < h;

Check y1 < f(x1) (if yes) there is a hit

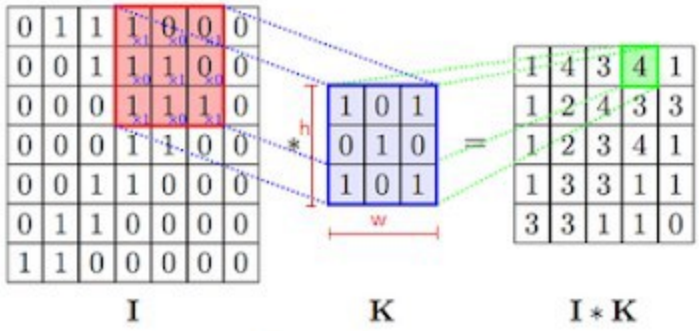
//this a random point generator

```
#include <iostream>
#include <cmath>
#include   <cstdlib>
```

```
double randMToN(double M, double N)
{
   return M + (rand() / ( RAND_MAX / (N-M) ) ) ;
}
```

Easy for an AI apprentice:

Convolution



$$(I * K)_{xy} = \sum_{i=1}^{h} \sum_{j=1}^{w} K_{ij} \cdot I_{x+i-1,y+j-1}$$

Some examples of using convolutions:

https://medium.com/@er_95882/convolution-image-filters-cnns-and-examples-in-python-pytorch-bd3f3ac5df9c