

# Homework #12

(Due date: 14 December 2023)

## Objective :

Learn how to use **file input and output**, **strings**, and **functions**.

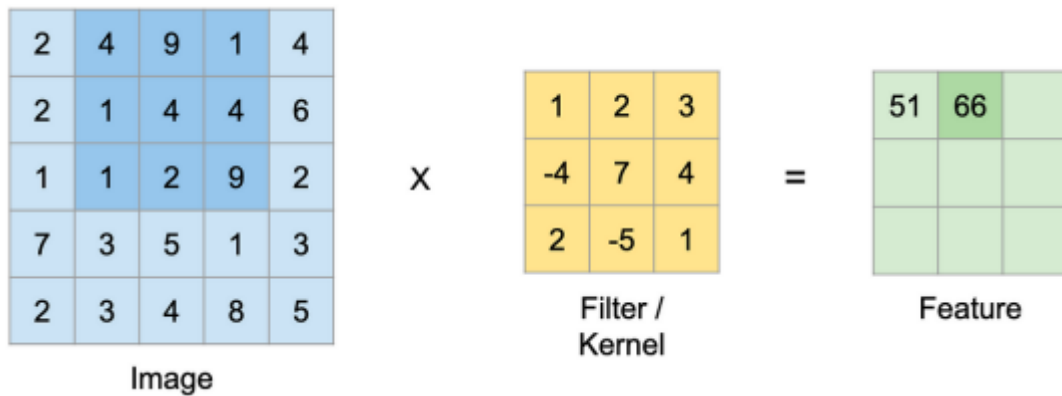
### 12.

Convolutional Neural Networks (CNNs) are a type of feedforward neural network that excels in machine learning tasks involving large images. A critical component of CNNs is the 2-D Convolution layer, which uses the weights in the kernel to perform convolution operations with the data input from the previous layer, and then passes the result through an activation function to produce the output for that layer.

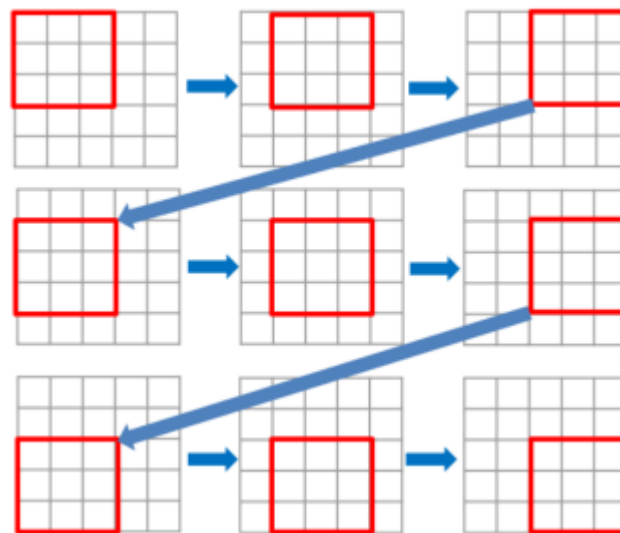
Please design a program to implement 2-D Convolution with the following requirements:

First, please write a function use `srand(5)` as the seed to generate a random number matrix (16x16) with values ranging from -10 to +10, and output it as "Matrix.txt". Additionally, output a bias matrix (14x14) where all elements are 1, named "Bias.txt". Print and output both matrices.

Second, write a function to use the previously generated random number matrix and bias, along with the provided "kernel.txt" file, to perform convolution operations. The process for matrix convolution operation is as follows:



Taking the above figure as an example, the calculation process of convolution operation is as follows:



$$2*1 + 4*2 + 9*3 + 2*(-4) + 1*7 + 4*4 + 1*2 + 1*(-5) + 2*1 = 51$$

$$4*1 + 9*2 + 1*3 + 1*(-4) + 4*7 + 4*4 + 1*2 + 2*(-5) + 9*1 = 66$$

$$9*1 + 1*2 + 4*3 + 4*(-4) + 4*7 + 6*4 + 2*2 + 9*(-5) + 2*1 = 20$$

$$2*1 + 1*2 + 4*3 + 1*(-4) + 1*7 + 2*4 + 7*2 + 3*(-5) + 5*1 = 31$$

...

$$2*1 + 9*2 + 2*3 + 5*(-4) + 1*7 + 3*4 + 4*2 + 8*(-5) + 5*1 = -2$$

After the operation, write a function add the bias to get a 14x14 matrix, print it out, and output it as "Conv.txt".

Process the output from point 2 through the ReLU activation function ( $\text{ReLU}(y) = \max(0, x)$ ), and print and output the result as "Result.txt".

Hint:

The files you need to submit are "Matrix.txt", "Conv.txt", "Result.txt", and the .c file for the question.

The output of the convolution operation should be rounded to the nearest integer. For example, 1.50 rounds up to 2, and 1.49 rounds down to 1.

ReLU is a function that zeroes out negative values, meaning it retains positive values as is and converts negative values to 0.

The names of the functions are generate() and convolutional( ) and relu( ) respectively.

Output :

```
Input :
  4   3   5  -1 -10   0  -3  -1   2 -10   4  -9   3   6  -4  -3
 -7  -3   0   1   7   2   3   1   6  -1   2  -7  -8   1   1  -1
  9   0   3   5  -8  -3  -7   4  -4  -4   4  -5   5   9 -10   5
 -4   2  -6  -5   1   9   3   9   0  -6   7   6  -2  -4   8  -9
  6   0  -1   0   3   1  -4   5   0  -2  -4  -8   5   5  -8 -10
  3 -10  -1   3  -5   0   6   9   3  -7  -5  -4  -6   7   8   8
 -10  9  -3  -5 -10  -4   1 -10   4   8  -5  -7   3   3  -1   9
  6  -9   5   2  -7   5  -4   9  -4   0  -4  -9   5  -4   8   2
 -8   1  -3  -8   6   7   1   6  -2  -9  -5  -1  -4   1   3  -3
 -1  -9  -4   3  -8  -6   1  -1  -7   4  -3   2   2   6   6   6
  7   7  -5   3  -3   8  -4   2  -2  -3  -2   9  -9   8  -8  -5
 -4   5   6   3  -1   2  -8  -2   4  -9   1 -10  -3   4   8   0
  0 -10   2  -6   5  -4  -2  -3   6   7   1   4  -4  -4  -2   6
 -8   0   5   9  -6  -4   6   4   1   3   1 -10  -5   8  -5   3
  1  -4   4   2   2   3 -10  -8   0 -10   0   8   8  -3   8  -5
 -4  -6  -6  -2  -6   6   7 -10  -6 -10   5  -1 -10   5  -1  -8

Bias :
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1

Kernel :
 0.330  0.330  0.330
 0.540  0.890  0.540
 0.330  0.330  0.330
```

Convolution :

2	5	4	3	-2	2	4	2	2	-8	-8	-6	1	3
2	3	2	0	-2	4	6	0	0	0	1	3	5	-2
3	-4	-5	3	5	12	9	2	-4	1	8	4	3	-2
-2	-6	-2	5	6	13	13	5	-6	-11	-7	3	7	-3
-6	-4	-5	-7	-3	8	13	6	-6	-15	-14	-2	11	16
0	-3	-11	-13	-9	3	1	7	2	-12	-14	-2	11	16
-6	-1	-6	-6	0	5	5	1	-6	-14	-13	-3	6	11
-8	-9	-7	2	5	10	4	-2	-15	-12	-7	-2	9	11
-10	-7	-6	-3	-3	5	-3	-6	-8	-2	-2	8	8	12
6	3	-1	2	-2	-2	-5	-5	-7	-2	-1	2	7	5
7	8	5	4	-2	-7	-4	1	-2	-3	-9	-4	1	9
-5	3	5	1	-5	-5	2	9	11	0	-6	-8	-3	6
-3	6	11	-1	-5	-2	3	3	6	1	-4	-2	4	2
-6	3	4	4	0	-8	-11	-15	-10	-4	6	7	4	4

Result :

2	5	4	3	0	2	4	2	2	0	0	0	1	3
2	3	2	0	0	4	6	0	0	0	1	3	5	0
3	0	0	3	5	12	9	2	0	1	8	4	3	0
0	0	0	5	6	13	13	5	0	0	0	3	7	0
0	0	0	0	0	8	13	6	0	0	0	0	11	16
0	0	0	0	0	3	1	7	2	0	0	0	11	16
0	0	0	0	0	5	5	1	0	0	0	0	6	11
0	0	0	2	5	10	4	0	0	0	0	0	9	11
0	0	0	0	0	5	0	0	0	0	0	8	8	12
6	3	0	2	0	0	0	0	0	0	0	2	7	5
7	8	5	4	0	0	0	1	0	0	0	0	1	9
0	3	5	1	0	0	2	9	11	0	0	0	0	6
0	6	11	0	0	0	3	3	6	1	0	0	4	2
0	3	4	4	0	0	0	0	0	0	6	7	4	4

**Please note:: users of visual studio add the following code to your program's beginning**

```
2  #pragma warning(disable:4996)
```

繳交格式及規定：

程式重點地方請加註解,給分也會酌量參考註解。

請繳交 .c 檔即可。

c 檔的檔名一律統一,以學號為檔名壓縮成一個以學號為名的壓縮檔  
上傳,

上傳請一律繳交壓縮檔。

Example:

若學號為 B123456789, 則.c/.cpp 檔名為 B123456789.c ,

而壓縮檔名為 B123456789.rar 。

繳交期限 2023.12.14 (四) 09:00 分之前,逾期一律不收,

無輸入輸出及逾期者一律以 0 分計算。

作業請上傳中山網路大學 網大上傳方式:

1. 點選要繳交的作業,選擇「進行作業」。



2. 依照流程上傳檔案。

