***Course Title***

***Course Code***

***Lab Journal***

****

|  |
| --- |
| **STUDENT NAME: “MUHAMMAD HAIDER AKBAR ZUBAIRI”** |
| **&&** |
| **“MUHAMMAD SHERAZ”** |
|  |
| **ENROLMENT NO. : 01-134202-085, 01-134202-047** |
| **CLASS AND SECTION BS-CS-2-B** |

**Submission Date**

UML DIAGRAM:

|  |
| --- |
| Matrix |
| - row\_no: int  - column\_no: int |
| + Matrix()  + Matrix(n: int,m: int)  + get\_rows\_no():int  + set\_rows\_no(input: int): void  + get\_column\_no():int  + set\_column\_no( input1: int): void  + read\_from\_file(file\_name: char[],row: int&, col: int&):void  + getting\_values\_from\_file(a: int\*\*, R\_limit: int&, C\_limit: int&): void  + getting\_values\_randomly(list: int\*\*, R\_limit: int&, C\_limit: int&): void  + Printing\_matrix(list: int\*\*, R\_limit: int&, C\_limit: int&): void  + add(matrix\_array1: int\*\*, matrix\_array2: int\*\*, R\_limit: int&, C\_limit: int&): void  + sub\_marix(matrix\_array1: int\*\*, matrix\_array2: int\*\*, R\_limit: int&, C\_limit: int&): void  +mult\_matrix: (int\*\* matrix\_array1, int\*\*, R\_limit: int&, C\_limit: int&): void  + transpose(matrix1: int\*\*, R\_limit: int, C\_limit: int): void  + Menu\_driven\_funcion(int\*\* matrix\_array1, int\*\* matrix\_array2, int& R\_limit, int& C\_limit): void |

* In this program, user is asked to enter the name of the file; //detail.txt
* From which it will read Column and Row Number respectively
* Two arrays are initialized name as (matrix \_array1, matrix\_array2)
* matrix \_array1 is generated Randomly.
* matrix\_array2 is constructed by reading from file //Num\_file.txt
* matrix\_list is generated Randomly.
* For the Function of ADD, Subtract, Transpose matrices (matrix \_array1, matrix\_array2) are used whereas for Multiplication (matrix \_array1, matrix\_list) is used.
* No. of rows and no. of column for the matrix (matrix\_list) are again accessed from the user for generating matrix.

*Note: keeping in view the fact for Matrix Multiplication You can only multiply two matrices if their dimensions are compatible, which means the number of columns in the first matrix is the same as the number of rows in the second matrix.*