

BIL105-E

Intr to Sci&Eng Comp

CRN: 22596

III.Homework

Instructor:

Yusuf YASLAN

Student's Name and Number:

Tuğrul YATAĞAN

040100117

Date: 28.05.2012

I. Introduction

Two different matrices will be read from text files (input1.txt, input2.txt) as a command line argument and they will be stored in two dimensional arrays (matrix1, matrix2) and one dimensional arrays (array1, array2). Our aim is to obtain the matrix multiplication using two dimensional and one dimensional arrays.

II. Development and Runtime environment

The project was developed on GNU/Linux **Ubuntu 12.04** distribution operation system. **GCC** (g++) compiler was used for compiling and **Geany** was used as integrated development environment editor.

Program compiles and then run with three input parameters. Program works on *main* function with;

- ***void read_file***: Reads elements of a matrix from a in_file and stores the elements into one dimensional array and two dimensional matrix parameters.
- ***void write_file***: Writes the elements of a matrix to out_file.
- ***void print_matrix***: Prints the elements of a matrix to the screen.
- ***void print_array***: Prints the elements of a matrix (matrix is represented as an array) to the screen.
- ***void multip***: Function that multiplies two dimensional arrays.
- ***void multip_array***: Multiplies one dimensional arrays array1 and array2 array and stores the results at one dimensional array array_result.

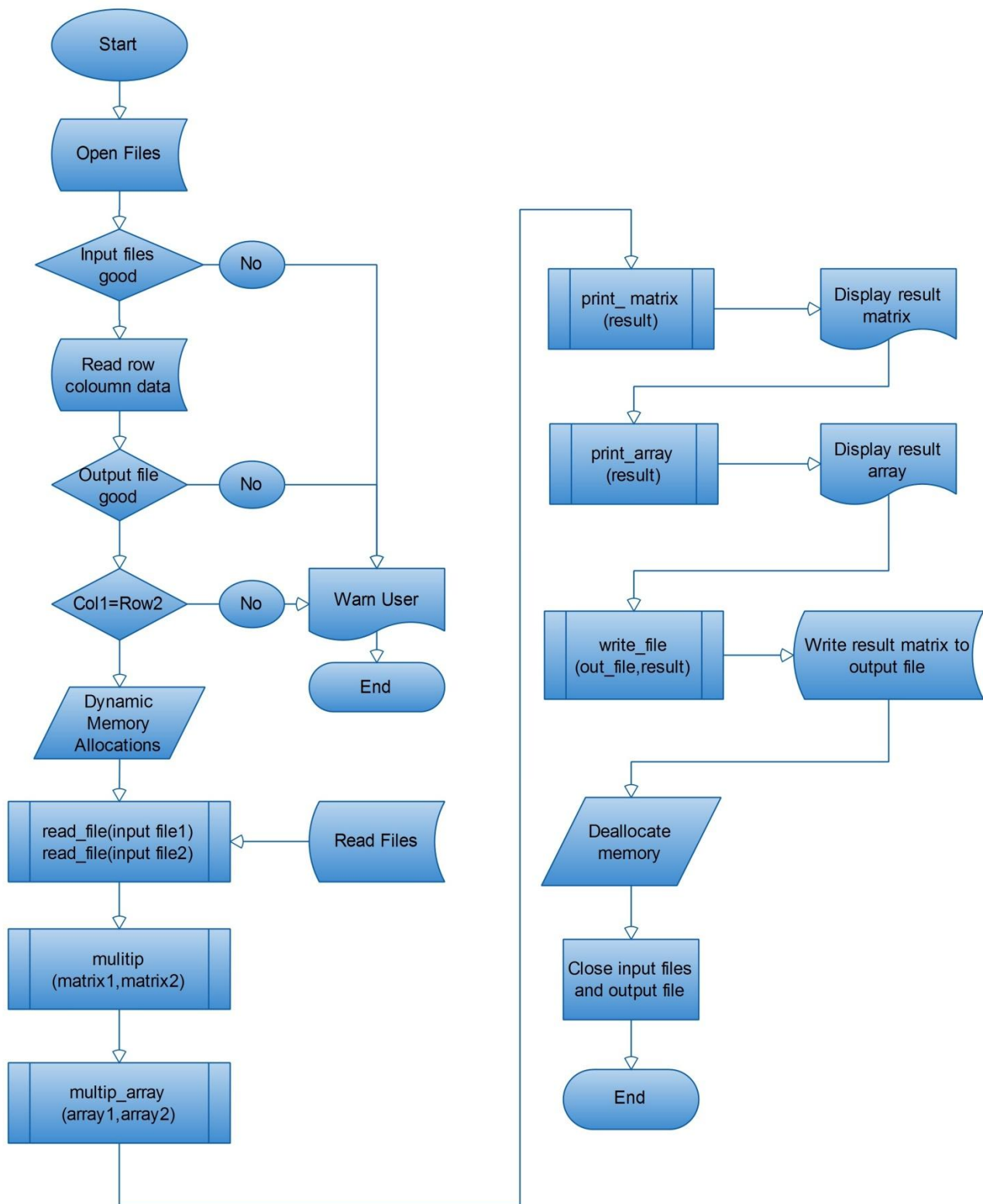
functions.

Program includes *iostream*, *fstream*(for reading/writing text files), *io manip*(editing output files) standard C++ libraries.

III. Variables

row1, *col1*, *row2* and *col2* integer variables which are firstly taken from input and these are used for limit matrix and array size. ***matrix1***, ***matrix2***, ***result***, ***array1***, ***array2*** and ***result_array*** double variables which are generated dynamically. These are dynamically allocated according to row and column information from input. These arrays and matrices passed to functions as pointer

IV. Program Flowchart



V. Conclusion

The program is compiled without any compiling warning. Necessary structures are applied on program. It is consistent and it runs without any runtime error. Program responds user's requirements. The project reached aimed target.