

# CS262, Project 2:

## Spreadsheet

Due: **Sunday, April 10, at 11:59 pm ET**

### Overview:

In this project you will write a C program to perform basic math operations in an  $5 \times 5$  spreadsheet.

In a spreadsheet, data is entered into a 2D array of rows and columns. In Excel, for example, rows are labeled with numbers [1, 2, 3, ...] and columns are labeled with letters [A, B, C, ...]. A cell is the place where data is entered. To indicate a specific cell, its coordinate is entered, which is the intersection point of a **column** with a **row**, e.g., B7. There are different operations that can be performed on the spreadsheet. Among the operations for numeric type data are addition and multiplication, these operations can make use of the *range* operator, which is represented by a colon (:) and is used to indicate a *consecutive group of cells*.

### Description:

The C program simulates a small spreadsheet that performs operations with numeric type data (double). The workspace is a  $5 \times 5$  table, where a cell is indicated by the intersection of a **column** Letter [A, B, C, D] with a **row** Number [1, 2, 3, 4, 5].

Operations are performed using reserved words: ADD, MUL, DEL, INS followed by *parentheses* and ending the operation with a semicolon (;). The cell, or range of cells, where the operation is performed is indicated within the *parentheses*. Empty parentheses indicate that the operation is performed on all data in the spreadsheet. If a range is indicated, for it to be valid the cells must meet:

1. Belong to the same row (start column < end column) or
2. Belong to the same column (start row < end row)
3. A cell has the format **LN**, where L=[A, B, C, D] and N=[1, 2, 3, 4, 5].

Notes: Constraint 3 also applies when the argument is a single cell.

The program is not case sensitive (toupper() could be used to process the input).

In this assignment, not all operations can be performed with all 3 types of arguments (cell, range of cells or empty parentheses). e.g., for **addition** and **multiplication**, only empty parentheses or a range of cells can be applied, while for **delete** and **insert**, all 3 types of arguments can be applied. The **delete** operation replaces the contents of the cell(s) with the value 0.00. The **insert** operation inserts a value in the indicated cell(s).

Operation	(cell)	(cell:cell)	( )	Examples		
ADD		✓	✓	ADD (A2:C2) ;		ADD ( ) ;
MUL		✓	✓	MUL (E1:E3) ;		MUL ( ) ;
DEL	✓	✓	✓	DEL (B4:D4) ;	DEL (A2) ;	DEL ( ) ;
INS	✓	✓	✓	INS (D1:D5) 5.3 ;	INS (C3) 3 ;	INS ( ) 1.1 ;

**Table 1:** summary of operations in the spreadsheet

When an operation is indicated, there are 3 different categories of errors:

1. **Syntax Error:** Err msg is “Invalid operation”
2. **Wrong Range:** The specifications (1,2,3) for range were not meet. Err msg is “Wrong range”
3. **Invalid Cell:** A single cell specified in parentheses is not valid. Err msg is “Invalid cell”

## Detailed requirements:

When starting the program, the spreadsheet should be displayed with the initial values (0.00) along with the column and row labeling. Each column is separated by a tab and each row is printed on a new line. The values in the table are rounded to two decimal places.

*An example of displaying the spreadsheet at startup is as follows:*

	A	B	C	D	E
1	0.00	0.00	0.00	0.00	0.00
2	0.00	0.00	0.00	0.00	0.00
3	0.00	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	0.00	0.00
5	0.00	0.00	0.00	0.00	0.00

- Whenever the spreadsheet is displayed, *the symbol for data entry (>>)* is placed below it.
- Each time “Enter Key” is pressed *the symbol for data entry (>>)*, the result of the operation is displayed and below it (in a new line), the symbol >> is displayed again. Example:

```
>>operation
result
>>
```

- When the INS or DEL operations are used, the `result` is the updated spreadsheet (displayed).
- If the indicated operation generates an error, then, `result` is the corresponding **Err msg**.
- Write “quit” just after >> and press “Enter” to exit the program

## Makefile:

Create a *Makefile* to compile your program and to remove the executable.

Use the same template of the *Makefile* specified on Lab7 assignment

The name of the source file will be `p2_<username>_<labsection>.c`

## Submitting:

1. On zeus, create a directory named `p2_<username>_<labsection>`.  
Copy your source file and *Makefile* to this directory.
2. Create a typescript with the following content:
  - a. Show that you are on zeus,
  - b. Show a listing of your directory
  - c. Show your source code
  - d. Compile the code using the *Makefile*.
  - e. Run your program using at least 3 examples listed in Table 1.  
In the same run, enter values to display the 3 different **Err msgs**.
3. Be sure your directory ONLY contains the *source file*, *typescript* and *Makefile*
4. Change to the parent directory and create a tarfile of your project directory.  
Name this tarfile `p2_<username>_<labsection>.tar`
5. Submit this tarfile to Blackboard no later than Due Date.

**Congratulations! You have completed your Project**

