EL2208 Problem Solving with C Lab. Lab #11

Introduction to 2-dimensional arrays and functions.

Use praktikum.ee.itb.ac.id as usual.

This lab will serve as an introduction to 2-dimensional arrays and functions. You will write a program which reads temperature data each day for four weeks, computes the average for each week, prints the daily temperatures followed by the average for that week.

The requirements for this lab are that there will be a main program containing the variable declarations and the function calls plus the three functions which do the bulk of the work. The first function will read the input data from a file called **lab11.in**. The data is formatted in the following fashion:

```
3 7   <---# of rows and # of columns (weeks and days, respectively)
45.9  <---array location [0][0]
23.1  <---array location [0][1]
56.7  <---array location [0][2]
89.3  <---array location [1][0]
  .              .
  .              .
  .              .
```

The first number in the data file is the number of weeks of temperatures followed by the number of days in a week. Read both of those sentinels into integer variables and then read the temperature data into an array. You can assume for declaration purposes, that the temperature array will not contain more than 10 weeks worth of data and will contain 7 columns per week. Use the following prototype:

void read_input(double temp[][7], int *weeks, int *days);

There is a sample input file in the **labs** subdirectory of **praktikum.ee.itb.ac.id** accessible via anonymous FTP.

The next function will compute the average temperature for each week and put that into a 1-dimensional array. The row that the average is stored into will be the same as the row for the week of temperatures being accessed; i.e. the average of the temperatures of week 2 will go into row 2 of the average array. Use the following prototype:

void compute_average(double temp[][7], int weeks, int days, double average[]);

The last function will print the output to a file called **lab11.out** in the following format:

```
Sunday  Monday  Tuesday  Wednesday  Thursday  Friday  Saturday  Average
 45.9     23.1    56.7      89.3      77.1      54.2     44.6      55.8
 50.1     67.9    77.6      71.2      82.1      88.9     79.8      73.9
 66.8     56.9    77.4      80.2      70.5      66.7     75.2      70.5
```

Print all floating-point data to one decimal place.

When the code is working correctly show the lab instructor the completed code and accompanying output.

Submit the code using usual perl submitter.