

Exercise 1 – Temperatures

Date Due: 5:40PM, February 5th, 2015

Write a program that uses a structure (CityTempInfo) to store the following data about temperatures:

- cityName
- TempInfo

Your program should read the data from an input file which has the following format:

```
4
12
Miami,59.6,60.5,64.0,67.6,72.0,75.2,76.5,76.5,75.7,72.2,67.5,62.2
Miami,76.5,77.7,80.7,83.8,87.2,89.5,90.9,90.6,89.0,85.4,81.2,77.5
Key West,65.2,65.7,68.8,72.1,75.9,78.7,79.6,79.2,78.5,75.7,71.9,67.3
Key West,75.3,75.9,78.8,81.9,85.4,88.1,89.4,89.5,88.2,84.7,80.6,76.7
Jacksonville,41.9,44.3,49.8,54.6,62.5,69.4,72.4,72.2,69.4,59.7,50.8,44.1
Jacksonville,64.2,67.3,73.4,78.6,84.3,88.7,90.8,89.4,86.1,79.1,72.5,65.8
Pensacola,42.7,45.4,51.7,57.6,65.8,72.1,74.5,74.2,70.4,59.6,51.1,44.7
Pensacola,61.2,64.4,70.2,76.2,83.4,89.0,90.7,90.1,87.0,79.3,70.3,63.4
```

- The first number (4) is the number of cities
- The second number (12) is the number of months.
- Followed by City name and 12 **low Fahrenheit** temperatures, one for each month
- Followed by City name and 12 **high Fahrenheit** temperatures, one for each month

Note: There are two rows for each city, one with low temperatures and one with high temperatures.

Use this information to dynamically allocate an array of CityTempInfo structures. For each city structure, dynamically allocate an array TempInfo structures with members that can hold high and low temperatures as well as the temperature scale:

```
struct TemperatureInfo
{
    char scale; // F for Fahrenheit, or C for Celsius
    double low;
    double high;
    string monthName;
};
```

The program should have a menu that lets the user perform the following operations:

```
Welcome to COP 2335 Temperature Wizard Program. Please select one of the
following cities:
1. Miami
2. Key West
3. Jacksonville
4. Pensacola
5. Exit
```

Note: Choosing "Exit" terminates the program.

When the user selects a city, they should be prompted to select one of the following menu options:

Please choose one of the following options:

1. Degrees Celsius
2. Degrees Fahrenheit
3. Exit this menu

Note: Choosing “Exit this menu” does not cause the program to terminate, it simply displays the previous menu for the user to choose another city, accomplish this with nested loops.

If the user chooses option 1 or option 2, then the program needs to present the following menu:

What would you like to do?

1. View low and high temperatures
2. Find out what is the warmest month
4. Find out what is the coldest month
5. Exit this menu

Note: Choosing “Exit this menu” does not cause the program to terminate, it simply displays the previous menu for the user to choose another temperature choice, accomplish this with nested loops.

Below is an example of a neatly formatted output for Pensacola low and high temperatures in degrees Fahrenheit:

Pensacola		
Month	Low	High
Jan	42.7 F	61.2 F
Feb	45.4 F	64.4 F
Mar	51.7 F	70.2 F
Apr	57.6 F	76.2 F
May	65.8 F	83.4 F
Jun	72.1 F	89.0 F
Jul	74.5 F	90.7 F
Aug	74.2 F	90.1 F
Sept	70.4 F	87.0 F
Oct	59.6 F	79.3 F
Nov	51.1 F	70.3 F
Dec	44.7 F	63.4 F

To accomplish this, you will need to use manipulators from the <iomanip> header. Should the user select the option to view the coldest and warmest month, the program needs to display the **month name** – i.e. **January**.

The temperature conversion formulas are:

Fahrenheit to Celsius	Celsius to Fahrenheit
$^{\circ}\text{C} = \frac{5}{9} * (^{\circ}\text{F} - 32)$	$^{\circ}\text{F} = \frac{9}{5} * ^{\circ}\text{C} + 32$

Your program needs to be modular (use functions). **Use defensive programming practices.** Pass the data by value, reference, and/or pointers, as you deem appropriate. Since you will be allocating memory dynamically, be sure to de-allocate memory accordingly.....**no memory leaks!** Submit your work as **Ex.01-LastNameFirstNameInitial.cpp** - i.e. **Ex.01-ZejnilovicA.cpp** and upload it using the "Attachments" link/button