CST 180 C++ Programming
Objects Program Assignment (30 pts)

HAIL
SLEET
HAIL
SLEET
SORRY, FOLKS... I GOT NOTHIN'...
FOG
SLEET
FLOOD
SNOW
HAIL
FOG
FLOOD
WEATHER 5
U.S. BRACES FOR MORE WACKY WEATHER.

Purpose

To build a C++ program using classes and the standard string class.

Specifications

Write a C++ program that will decode raw weather data in text form. Your program should read a file (file name weatherObs.txt) and decode the information on each line and write it for the user in a decoded, narrative format.

The input file will appear as:

```
MBS,35,31,NW,12,3009,R
BAX,33,29,N,17,3006,S
MOP,43,41,NE,7,3004,DR
LAN,35,29,SE,8,3000,PC
AMN,41,38,SW,10,2999,CLD
FNT,39,36,S,4,2994,CLR
```

Your output for these data should be decoded to form similar to this example:

At Saginaw, the temperature is 35 degrees Fahrenheit (2 degrees Celsius), the dew point is 31 degrees Fahrenheit, winds are from the northwest at 14 miles per hour, pressure is 30.09 inches of mercury, and it is raining.

At Bad Axe, the temperature is 33 degrees Fahrenheit (1 degrees Celsius), the dew point is 29 degrees Fahrenheit, winds are from the north at 20 miles per hour, pressure is 30.06 inches of mercury, and it is snowing.

... and so on for the remaining reports.

Your program should include a driver program to read one line of coded weather data at a time as well as a class designed to store and manage information related to one weather observation. Your class should store all of the fields coming in from the data file defined with an appropriate data type for each. The class should include constructor(s) and set/get functions, as required.

Extraction of the information from the raw data should be defined in the class. File processing should be included in your main driver function. As you read a new coded weather observation, format it from the raw form to extract all of the various fields in the appropriative data type. Invoke a function on your weather observation object to write the narrative information for the given observation. Then, read the next line in the file and process it the same way.

The main task to be performed as a function in your class will be to extract, or "parse", the data from the input string, decode it, and perform some conversions before you can store the information in the member variables of your class object. Utilize the standard C++ string class for string processing. Be sure your data members that are to be stored as character strings use the string class. This implies that class *composition* will be included by having your weather observation class contain objects of the standard string class.

Details on each data field are below:

Field	Description
Station code	Always 3 characters. Only a predefined list of regional weather stations will be used. Codes for these are: AMN Alma BAX Bad Axe FNT Flint LAN Lansing MBS Saginaw MOP Mount Pleasant
Temperature	Provided in degrees Fahrenheit. Your output should include the equivalent Celsius temperature. To convert,subtract 32 from the Fahrenheit temperature and multiply the result times 5/9.
Wind direction	Using an eight-point compass reference. You will need to decode to write the descriptive form to output: N north NE northeast E east SE southeast S south SW southwest W west NW northwest
Wind speed	Provided in knots, or nautical miles per hour. This must be converted to miles per hour for output. To convert, multiply knots by 1.151.
Pressure	Provided in coded form with the unit of "inches of mercury". A coded value of 3005 implies 30.05 inches, a coded value of 2997 implies 29.97. Pressure is always reported to two decimal places even if they are .00.
Weather	A code is included describing the current weather observed at the station. It will need to be decoded for output: CLD skies are cloudy PC skies are partly cloudy CLR skies are clear R it is raining S it is snowing DR drizzle is reported

	T	a thunderstorm is reported

Design Requirement

Create the abstract data type (like a structure) for the "weather observation" class you are creating. Clearly indicate the "data members" (including data types), or attributes of one weather observation. Also, indicate a list of the functions that are required within the scope of this programming assignment. Finally, include the operations on the string class necessary to extract the station code and the temperature from one line of raw coded data.

Final Deliverables

Deliver a single document with the following:

- Cover page with assignment name, student name, and list of attachments
- UML Diagram of your weather class
- Printed copy of source code files
- Printed copy of output decoding all records in the data file provided.

Upload the document **AND** the .cpp source code file for grading into the appropriate dropbox.