Student Name: Siyuan Chen

Aim of the Project:

1. This project can allow users to get specific information from the system.
   Ex:
   1)Average temperature, relative humidity, absolute humidity of specific month;
   2) Get the temperature, and relative humidity at specific date and time;
   3)Highest temperature, relative humidity, absolute humidity of specific month;
   4)Display all the temperature, relative humidity, absolute humidity higher than the average data of that month of year.
2. This project helps us strengthen the practical use of C++ IO, how to split data with delimiter, store information in vector, convert string to integer, how to generate a menu option.

User Guide:

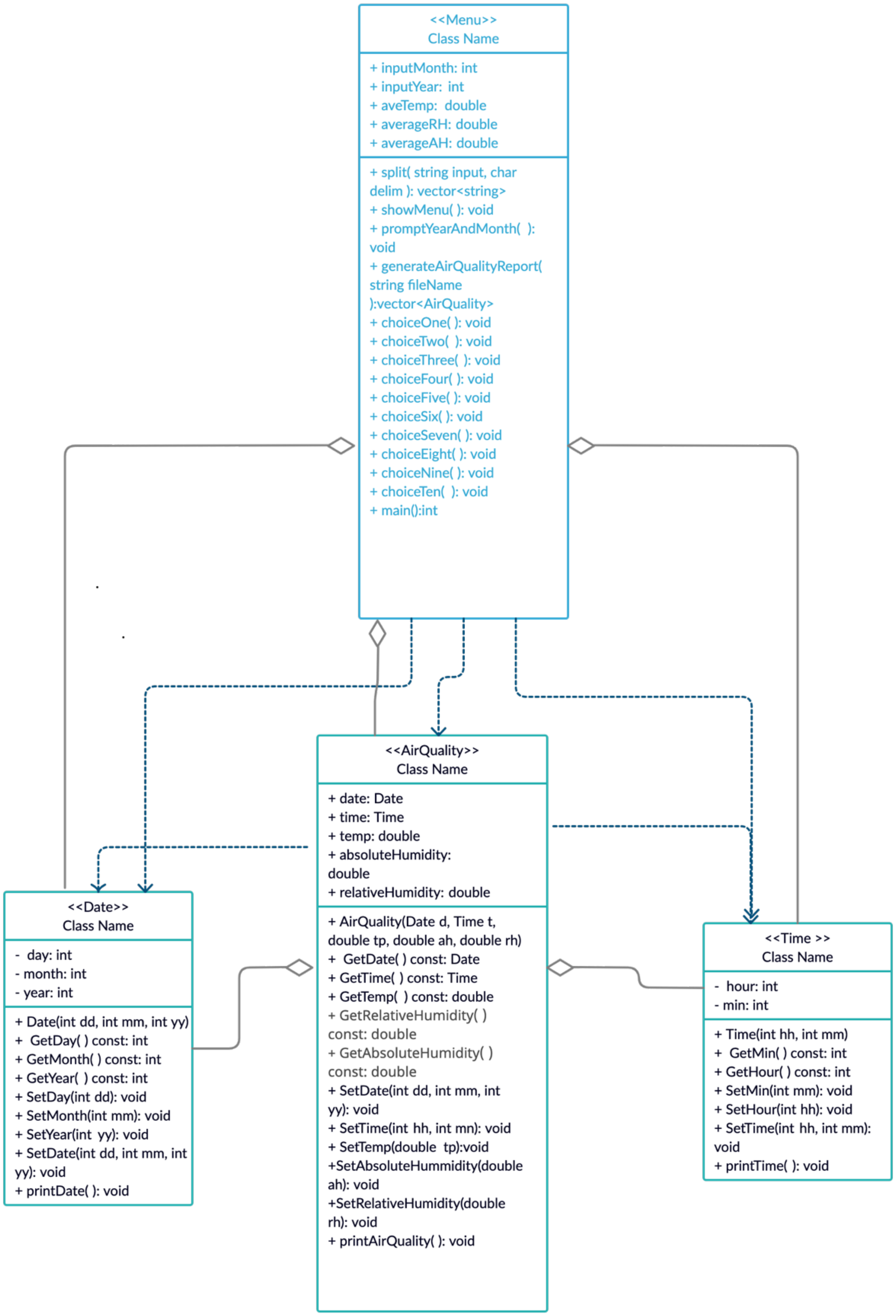
1) Run Menu Class to get all the information;
2) Select one choice from the menu options. You could input 11 to quit the system;
3) Select Choice 1 or 2 or 3, then input month, ex:03, 04, 05…
4) Select Choice 4, then input year:2004 or 2005,
   if input 2004, then input month from 3 to 12 ex:03, then input day ex:10, then input hour ex:18, then input minute ex:00.
   if input 2005, then input month from 1 to 4 ex:03, then input day ex:10, then input hour ex:18, then input minute ex:00.
5) Select Choice 5,6,7. Input valid month ex:10
6) Select Choice 8,9,10. Input 2004 or 2005, then input valid month, then it will display the result.

Project Description with Purpose:

1. Time.h
   Describe each methods and fields of Time including hour and minute
2. Time.cpp
   Implement each method as follows:
   Constructor, Set time, set hour, set minute, GetHour, GetMin, print time.
3. Date.h
   Describe each methods and fields of Date including year, month and day.
4. Date.cpp
   Implement each method as follows:
   Constructor, Setday, setmonth, set year, GetDay, GetMouth, GetYear, printdate.
5. AirQuality.h
   Describe each methods and fields of date, time, temperature, relative humidity and absolute humidity.
6. AirQuality.cpp
   Constructor, GetDate, GetTime, GetTemp, GetAbsoluteHumidity, GetRelativeHumidity, SetDate, SetTime, GetTemp, GetAbsoluteHumidity, GetRelativeHumidity

7. Menu.cpp
    1. vector<string> split(string input, char delim): helper method to split string with delimiter
    2. void showmenu(): helper method to display the menu options
    3. promptYearAndMonth(): to prompt the user to input year and month
    4. Read csv file, get each line and store in a vector
    5. Separate each line with comma and get each column of each line, then store into a vector
    6. Separate date column with slash, then set each month, day, year to a date instance
    7. Separate time column with comma, then set each hour, minute to a time instance
    8. Get each temperature, relative humidity and absolute humidity
    9. Initialize a AirQuality instance with each field
    10. Put the AirQuality instance into the airQualityReport vector
    11. Then generate a menu options, the customer can continuously choose 1 – 11 from the menu
    12. Choice 1 part is to prompt user to input a month, then calculate and output the average temperature.
    13. Choice 2 part is to prompt user to input a month, then calculate and output the average relative humidity.
    14. Choice 3 part is to prompt user to input a month, then calculate and output the average absolute humidity.
    15. Choice 4 part is to prompt user to input a specific year, month and time, then output that specific time’s temperature and relative humidity.
    16. Choice 5 part is to prompt user to get the highest temperature for that month.
    17. Choice 6 part is to prompt user to get the highest relative humidity for that month.
    18. Choice 7 part is to prompt user to get the highest absolute humidity for that month.
    19. Choice 8 part is to prompt user to display specific year specific month’s temperature higher than average temperature.
    20. Choice 9 part is to prompt user to display specific year specific month’s relative humidity higher than average relative humidity.
    21. Choice 10 part is to prompt user to display specific year specific month’s absolute humidity higher than average absolute humidity
    22. Choice 11 part is to quit the system.
    23. Main part is to run the system.

Class Diagram



Algorithm

1) Vector to store each line of csv file.
2) Vector to store each column of each line
3) Vector to store date column
4) Vector to store time column

Appropriate Source Code listing:

The following classes are from Source Code

1. Time.h and Time.cpp
2. Date.h and Date.cpp

Test Plan and Test Results

Date class

Test case No	Test case	Input	Output	Pass/Fail
1	Constructor: Date()	Date testDate3 ()	1 1 2000	Pass
2	Setter: SetDate(int dd, int mm, int yy)	testDate1.SetDate(10, 10, 2004)	10 10 2004	Pass
3	Setter: SetDay(int dd)	testDate2. SetDay (10)	10	Pass
4	Setter: SetMonth(int mm)	testDate2.SetMonth (10)	10	Pass
5	Setter: SetYear(int yy)	testDate2. SetYear (2004)	2004	Pass
6	Getter: GetDay()	testDate2. GetDay ()	10	Pass
7	Getter: GetMonth()	testDate2. GetMonth ()	10	Pass
8	Getter: GetYear()	testDate2. GetYear ()	2004	Pass
9	printDate()	testDate2. printDate()	10 10 2004	Pass

Time class

Test case No	Test case	Input	Output	Pass/Fail
1	Constructor: Time()	Time testTime3 ()	10:10	Pass
2	Setter: SetTime(int hh, int mm)	testTime1.SetDate(10, 10)	10:10	Pass
3	Setter: SetHour(int hh)	testTime 2. SetHour (10)	10	Pass
4	Setter: SetMinute(int mm)	testDate2. SetMinute (10)	10	Pass

6	Getter: GetHour()	testDate2. GetHour ()	10	Pass
7	Getter: GetMinute()	testDate2. GetMinute ()	10	Pass
9	printTime()	testDate2. printTime()	10: 10	Pass

AirQuality Class

Test case No	Test case	Input	Output	Pass/Fail
1	Constructor: AirQuality ()	AirQuality testAQ3 ()	Date:112001, Time:10:10,10, 20, 30	Pass
2	Setter: SetTime(int hh, int mm)	testAQ1. SetTime (20, 20)	20:20	Pass
3	Setter: SetDate(int dd, int mm, int yy)	testAQ1.SetDate(10, 10,2004)	10 10 2004	Pass
4	Setter: SetTemp(double tp)	testAQ1. SetTemp (18.8)	10	Pass
5	Setter: SetAbsoluteHumidity (double ah)	testAQ1. SetAbsoluteHumidity (5.5)	5.5	Pass
6	Setter: SetRelativeHumidity (double rh)	testAQ1. SetRelativeHumidity (10.1)	10.1	Pass
7	Getter: GetDate()	testAQ1. GetDate ()	10 10 2004	Pass
8	Getter: GetTime()	testAQ1. GetHour ()	20:20	Pass
9	Getter: GetTemp()	testAQ1. GetTemp()	18.8	Pass
10	Getter: GetAbsoluteHumidity ()	testAQ1. GetAbsoluteHumidity ()	5.5	Pass
11	Getter: Get RelativeHumidity ()	testAQ1. RelativeHumidity ()	10.1	Pass
12	printAirQuality ()	testAQ3. testAQ3 ()	Date:112001, Time:10:10,10, 20, 30	Pass

Menu Class

Test case No	Test case	Input	Output	Pass/Fail
1	Vector of lines:	Lines[0]	9358	pass

	Size and header	Lines.size()	Date,Time,CO(GT),PT08.S1(CO), NMHC(GT),C6H6(GT),PT08.S2(NMHC), NOx(GT),PT08.S3(NOx),NO2(GT), PT08.S4(NO2),PT08.S5(O3),T,RH,AH,,	
2	Test oneDate.printDate()	oneDate.printDate()	All the date information	pass
3	Test oneTime.printTime()	oneTime.printTime()	All the time information	pass
4	eachTemp	String getTemp = splittedLine[12] Double eachTemp = stod(getTemp)	All the temp information	pass
5	Each RH	String getRH = splittedLine[13] Double eachRH = stod(getRH)	All the RHinformation	pass
6	Each AH	String getAH = splittedLine[14] Double eachAH = stod(getAH)	All the AH information	pass
7	April average temp	2004,04	16.80	pass
8	April average RH	2004,04	50.71	pass
9	April average AH	2004,04	0.92	pass
10	2004/03/10/20:00 temp and rh	2004/03/10/20:00	11.90, 54.00	pass
11	04 month highest temp	04	31.30	pass
12	04 month highest rh	04	82.40	pass
13	04 month highest ah	04	1.49	pass
14	Test the average temp of 03/2004, Test the total number of 03/2004’s tempature higher than average	2004 03	14.39 239	pass
15	Test the average rh of 03/2004, Test the total number of 03/2004’s rh higher than average	2004 03	50.17 285	pass
16	Test the average ah of 03/2004,	2004 03	0.79 255	pass

	Test the total number of 03/2004’s ah higher than average			

State assumptions if any clearly.

1) I assume that if the input choice, month and year is character then will prompt user to input again
2) I assume that if the input choice is not 1-11, month is not 1-12 and year is not 2004 or 2005, then will prompt user to input again
3) I assume that for choice 4, if the output is 0, then it means that the table does not have that data for input time.

Reflection:

1) It is important to know c++ IO to read file and write file.
2) We could use vector to store each line of file so that we could index each column.
3) We also can write a helper method to split each line and each column with specific delimiter.
4) Following the objected-oriented programming principle is essential which can make each method become clear and readable.