Date class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor initialises the data	m_day = 0, m_month = 0, m_year = 0	Passed
2	Test the mutator for Day member	it should change the value of the day in the date class	Passed
3	Test the accessor for Day member	For accessor GetDay() should return the value of m_day	Passed
4	Test the mutator for Month member	it should change the value of m_month	Passed
5	Test the accessor for Month member	return the value of m_month	Passed
6	Test the mutator for Year member	changes the value of m_year upon execution	Passed
7	Test the accessor for Year member	returns the value of m_year	Passed
8	Test the overloaded >> operator to check whether the Date class has received the inputted values by specifying ',' at the end of input	accept the input for three different value (day/month/year)	Passed
9	Test the overloaded << operator for the output of the Date class members	output three values (day month year)	Passed
10	Test the overloaded == operator to compare both object of Date class members	return a boolean value for comparing two Date object (TRUE or FALSE)	Passed
11	Test the overloaded != operator to compare both object of Date class members	return a boolean value for comparing two Date object (TRUE or FALSE)	Passed
12	Test the overloaded < operator to compare both object of Date class members	return a boolean value for comparing two Date object (TRUE or FALSE)	Passed
13	Test the overloaded > operator to compare both object of Date class members	return a boolean value for comparing two Date object (TRUE or FALSE)	Passed
14	Test the overloaded >= operator to compare both object of Date class members	return a boolean value for comparing two Date object (TRUE or FALSE)	Passed
15	Test the overloaded <= operator to compare both object of Date class members	return a boolean value for comparing two Date object (TRUE or FALSE)	Passed

Time class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor initialises the data	m_hours = 0, m_minutes = 0	Passed
2	Test the mutator for m_hours member	it should change the value m_hours	Passed
3	Test the accessor for m_hours member	it should return the value of m_hours	Passed
4	Test the mutator for m_minutes member	it should change the value of m_minutes	Passed
5	Test the accessor for m_minutes member	it should return the value of m_minutes	Passed
6	Test the overloaded >> operator to check whether the Result class has received the inputted values	accept the input for three different value (marks, date object and unit object)	Passed
7	Test the overloaded << operator for the output of the Time class members	output two values (m_hours and m_minutes) in this format HH:MM	Passed
8	Test the overloaded == operator to compare both object of Time class members	return a boolean value for comparing two Time object (TRUE or FALSE)	Passed
9	Test the overloaded != operator to compare both object of Time class members	return a boolean value for comparing two Time object (TRUE or FALSE)	Passed
10	Test the overloaded < operator to compare both object of Time class members	return a boolean value for comparing two Time object (TRUE or FALSE)	Passed
11	Test the overloaded > operator to compare both object of Time class members	return a boolean value for comparing two Time object (TRUE or FALSE)	Passed
12	Test the overloaded >= operator to compare both object of Time class members	return a boolean value for comparing two Time object (TRUE or FALSE)	Passed
13	Test the overloaded <= operator to compare both object of Time class members	return a boolean value for comparing two Time object (TRUE or FALSE)	Passed

Weather class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor initialises the data	m_wind_Speed = 0.0 , m_solar_radiation = 0.0 , m_temperature = 0.0. Use the << operator to output all members value	Passed
2	m_wind_speed member	it should change the value m_wind_speed	Passed
3	m_wind_speed member	it should return the value of m_wind_speed	Passed
4	Test the mutator for m_temperature member	it should change the value of m_temperature	Passed
5	m_temperature member	it should return the value of m_temperature	Passed
6	m_solar_radiation member	it should change the value of m_solar_radiation	Passed
7	m_solar_radiation member	it should return the value of m_solar_radiation	Passed
8	Test the overloaded >> operator to check whether the Weather class has received the inputted values	accept the input for three different value (wind speed, temperature and solar radiation) and set these three members with these values	Passed
9	Test the overloaded << operator to check whether the Weather class has output all three values	System will output all three member values	Passed

BST class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor initialises the data	root = NULL	Passed
2	Check whether this BST could insert data using insert method	A node will be created, BST will not be empty and a success message will be displayed for successful insert	Passed
3	Search the BST using the given datatype data	return true if found an identical object or return false if data is not found	Passed
4	Check whether this BST is empty()	it should change the value of m_minutes	Passed
5	Perform in order traversal in BST	Value of the data in BST will be printed in	Passed
6	Perform pre order traversal in BST	accept the input for three different value (marks, date object and unit object)	Passed
7	Perform post order traversal in BST	output two values (m_hours and m_minutes)	Passed
8	Destroy a node in BST	Destroy the node recursively on the root node	Passed

WindLogType class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor creates the object	WindLogType class object can be used	Passed
2	Test the mutator for m_date member	it should change the value m_date	Passed
3	Test the accessor for m_date member	it should return the value of m_date	Passed
4	Test the mutator for m_weather	it should change the value of m_weather	Passed
5	Test the accessor for m_weather	it should return the value of m_weather	Passed
6	Test the mutator for m_time member	it should change the value m_time	Passed
7	Test the accessor for m_time member	it should return the value of m_time	Passed
8	Test the overloaded >> operator to check whether the WindLogType class has received the inputted values	accept the input for three different object value (weather object, date, time object)	Passed

Utility class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor creates the object	Utility class object can be used	Passed
6	To retrieve a data from a row of csv by skipping the delimiter	It should output the data based on the position of the data separated by the delimiter starting from zero	Passed
7	To output the string from the month based on the given integer (1-12)	the string name of the month will be output	Passed

Data class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor creates the object	Data class object can be used	Passed
2	check whether populateData() is able to insert multiple files from data/met_index.txt	both m_datetime_tree and m_data_map should not be empty. Both function checkBinaryTreeIsEmpty() and checkMapIsEmpty() should return false	Passed
3	Check the existence of the data in a binary tree (m_date_tree) using month and year	Return true if there is any data with matching month and year else return false	Passed
4	Check the existence of the data in a binary tree (m_date_tree) using year	Return true if there is any data with matching year else return false	Passed
5	Calculate total wind speed for month in a year	return the total value of wind speed in a month based on the data uploaded	Passed
6	Calculate total solar radiation for month in a year	return the total value of solar radiation for a month in a year	Passed
7	Calculate total temperature for month in a year	return total temperature for a month in a year	Passed
8	Calculate total number of record for wind speed in a month	Return the total number of record of wind speed data in a month	Passed
9	Calculate total number of record for temperature in a month	Return the total number of record of temperature data in a month	Passed
10	Calculate total number of record for solar radiation in a month	Return the total number of record of solar radiation data in a month	Passed
11	Calculate average wind speed per month in a year	Return the average wind speed in a month	Passed
12	Calculate average temperature per month in a year	Return the average temperature in a month	Passed
13	Calculate average solar radiation per month in a year	Return the average solar radiation in a month	Passed
14	Calculate average wind speed per year	Return average wind speed per year	Passed
15	Calculate average temperature per year	Return average temperature per year	Passed
16	Calculate average wind speed, temperature and solar radiation per month in the given year	System will output a file with the given filename in any format with all the information	Passed
17	Calculate the highest solar radiation with the given date	return the highest solar radiation in a given date	Passed
18	Calculate total solar radiation for a year	return the total solar radiation in a year	Passed

Data class Test Plan

18	Print all timing in a specific date with the highest solar radiation	Timing with highest solar radiation in a specific date will all be printed	Passed
19	Check whether m_date_tree is empty	return true if it is empty else return false	Passed
20	Check whether m_date_map is empty	return true if it is empty else return false	Passed

Program class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor creates the object	Program class object will be created	Passed
2	check whether displayMenu() displays menu	A working menu will be displayed on the output	Passed
3	print the information for average wind speed and ambient air temperature per month	If there is no data for that month in the given year, "NO Data" will be printed else the name of the month followed by the data of average wind speed and average air temperature will be printed	Passed
4	print average wind speed and ambient air temperature in a year	System will print out average wind speed and air temperature per month in a year else will print "NO Data" if there is no data	Passed
5	Print total solar radiation per year	Print out solar radiation per month in a year, will print "NO Data" if there is no data	Passed
6	Print the highest solar radiation in a given date	Print all timing with highest solar radiation in given date	Passed
7	Execute the program	Main program will be executed	Passed

Vector class Test Plan

Test	Description	Expected Output	Status
1	Check that constructor creates the object	Vector class object will be created	Passed
2	search an element in vector class	system should output the element at the given position in integer	Passed
3	add an template datatype element into the back of the vector	No expected output, the object should exist in the vector by iterating through a loop	Passed
4	Get the number of element stored in vector	return the number of element stored in vector	Passed
5	Get the maximum number of element that can be stored in this vector	return the max number of element that can be stored in vector	Passed
6	check whether the vector is full	returns a boolean value if vector is full	Passed
7	resize the vector to a specific size	vector will be resize according to the number specified and this can be checked using getCount() function	Passed
8	create a number of empty element in vector according to number specified	vector will have a number of empty element in vector according to number specified, this can be checked using getCount() function	Passed
9	clear all element in vector	vector should not have any element	Passed