Tom Barabasz
CS 162 400 U2019
07/06/2019
Project1 Reflection

Program Requirements:

This program requires creation of a class Ant. The ant moves one space at a time according to Langton's Ant simulation rules. This program implements methods for moving the ant in East/West direction and North/South direction, as well as changing the color of the space the ant just departed and determining ant's direction of movement.

To give the user the option to start the simulation or to quit, the program uses the menu function to receive an integer input of 1 or 2 with a help of a "do-while" loop.

The user is required to enter number of columns and rows to create a board. A "do-while" loop ensures the number of rows and the number of columns is greater than 0.

The user is required to enter an integer for the number of steps the ant will take. Input validation method ensures user's input is a valid integer greater or equal to 0. The ant is allowed to take zero steps.

For extra credit, the program gives the user the option to start the ant at a random position or for the user to specify a starting position. If random start is chosen, the ant is placed at a random location which is less than the number of row and less than the number of columns to ensure it fits on the board. If the user opted to enter the starting position, a "do-while" loop ensures entered position is on the board.

The program dynamically creates a 2D array[columns][rows] and deletes it before it exits to free any dynamically allocated memory; therefore, preventing memory leaks. The array is initialized to a blank space character.

The ant moves specified number of times, one space at a time, according to the rules. Switch statement directs the steps according to current location space color and ant's orientation.

Once all steps have been taken, the user is given an option to start the simulation again with the help of a "do-while" loop. If users opts to play again, the program redirects to the start of the main function.

Step	Start x	Start y	Orientation	Color	# of Rows	# of Columns	Exp X	Exp Y	Exp Orient	Exp Color
0	0	0	W	_	5	5				
1	0	4	N	_	5	5	0	4	N	_
2	1	4	E	_	5	5	1	4	E	_
3	1	0	S	_	5	5	1	0	S	_
4	0	0	W	#	5	5	0	0	W	#
5	0	1	S	_	5	5	0	1	S	_
6	4	1	W	_	5	5	4	1	W	_
0	9	9	W	_	10	10				
1	9	8	N	_	10	10	9	8	N	_
2	0	8	W	_	10	10	0	8	W	_
3	0	9	S	_	10	10	0	9	S	_

Tom Barabasz
CS 162 400 U2019
07/06/2019
Project1 Reflection

4	9	9	W	#	10	10	9	9	W	#
5	9	0	S	_	10	10	9	0	S	_
6	8	0	W	_	10	10	8	0	W	_
0	4	0	W	_	5	5				
1	4	4	N	_	5	5	4	4	N	_
2	0	4	E	_	5	5	0	4	E	_
3	0	0	S	_	5	5	0	0	S	_
4	4	0	W	#	5	5	4	0	W	#
5	4	1	S	_	5	5	4	1	S	_
6	3	1	W	_	5	5	3	1	W	_

This project was quite challenging because its requirements required attention to detail. It was easy to make a mistake in specifying ant's moves. Sketching, tracing, and tracking ant's moves to test program's accuracy was the most challenging part. It would've been helpful to have a set of predetermined results to expect, which was not possible because of the countless possible scenarios.

Another challenge was how to implement classes and function. I was a Board class as a possibility, but in the end, I found it to be unnecessary. The requirements were achievable without it.