CIS 200 - Lab 0201

1. Problem Statement

Create a program that reviews arrays and provides practice in compiling and running program in Windows and Linux.

2. Requirements

2.1 Assumptions

- User will enter positive integers
- Array will only contain positive integers
- Array will contain 6 positive integers
- Array will be formatted in a 3x2 fashion
- Command line input/output only

2.2 Specifications

- Program will initialize 2 dimensional array with below data
    - Rows 3
    - Columns 2
    - { {3,2} {4,5} {2,2} }
- Immediately print formatted array with EVEN positive integers identified
    - User must then enter new values to fill the array
        - For valid positive integer
            - Insert integer into user defined array
        - For invalid input
            - Alert user and prompt for new input
    - Print formatted user array with EVEN positive integers identified
    - Loop filling array with new values until user wishes to terminate program.

3. Decomposition Diagram

- 2 Dimensional Array Testing Program
    - Input
        - User inputs positive integers by command line
        - User inputs if they would like to continue or not
        - Validate user input before inserting it
    - Process
        - Iterating through array to insert values
        - Iterating through array to check if values are EVEN
    - Output
        - Format and print array of positive integers
        - Alert user if input is invalid
        - Ask if user would like to continue or not

4. Test Strategy

- Valid Data
- Invalid Data

5. Test Plan Windows Version

***All test cases appear the same in table for Linux, validated with screenshots.

Test Strategy	#	Description	Input	Expected Output	Actual Output	Pass/Fail
Valid Data	1	Positive Integers > Array	1 … 6	Integer Inserted	{1,2},{3,4},{5,6}	Pass
Valid Data	2	Valid Choice	y	Enter Again		
Valid Data	3	Valid Choice	N	Exit Program		
Invalid Data	1	Negative Integer > Array	-1	Prompt to retry		

6. Initial Algorithm

1. Define Global Constants
    a. Max Rows of 3
    b. Max Columns of 2
2. Create Function *howManyEven*
    a. Parameters and Returns
        i. Parameter: 2 Dimensional Array *arrayIntValues*
        ii. Return integer of how many even integers are contained in array
    b. Iterate through first portion of array
    c. If integer read is even
        i. Iterate counter to be returned
    d. If integer read is odd
        i. Continue reading
3. Create Function *printArray*
    a. Parameters and Returns
        i. Parameter: 2 Dimensional Array *arrayIntValues*
        ii. Return void
    b. Iterate through first portion of array
        i. Print formatted row
        ii. Print formatted column
4. *MAIN* FUNCTION
    a. Create and Initialize 2 Dimensional array *arrayIntValues*
        i. { {3,2} {4,5} {2,2} }
    b. Create 2 holder variables
        i. One integer to hold integer input
        ii. One char to determine continue/exit
    c. Use *printArray* to show default array to user
    d. Use *howManyEven* to show the user how many even integers there are
    e. Do until user enters ‘n’ to stop program
        i. Alert user: “All future non-integer inputs will be rounded down.”
        ii. Prompt user to enter new integers into the command line
            1. If input is valid
                a. Insert input into array
            2. If input is invalid
                a. Alert user and attempt again
        iii. Use *printArray* to display new user generated array
        iv. Use *howManyEven* to display number of user generated even integers
        v. Prompt user: “Would you like to insert more positive integers?”

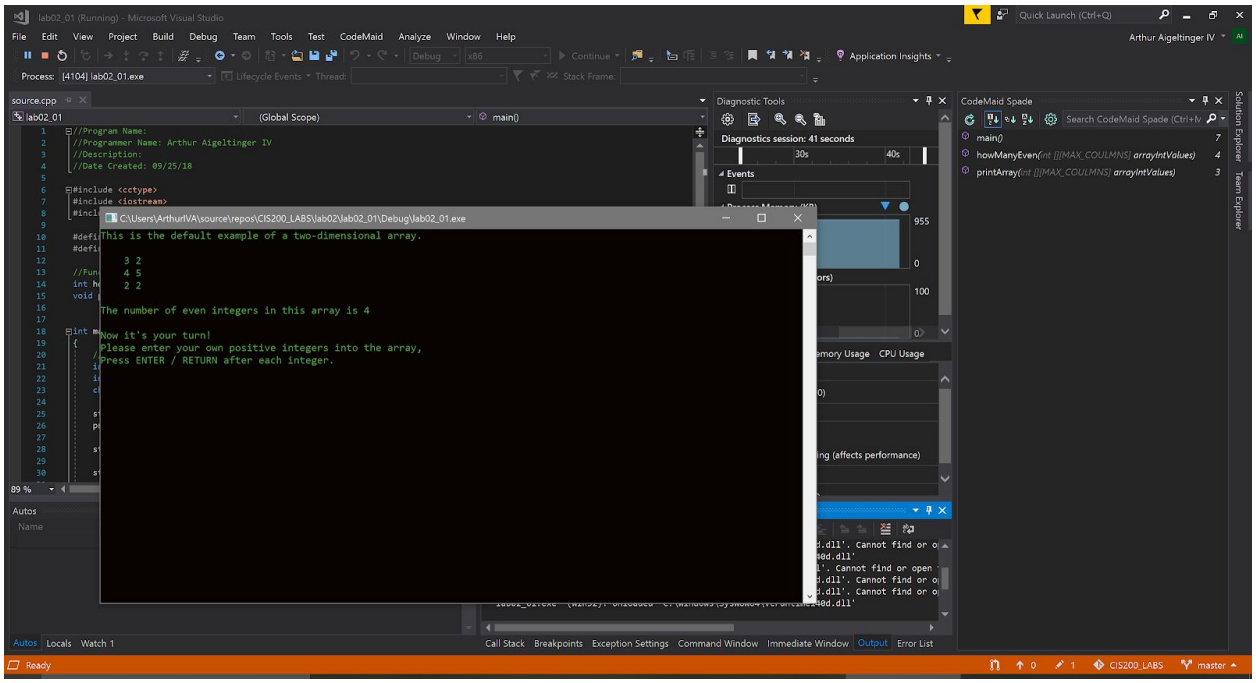
8. Code

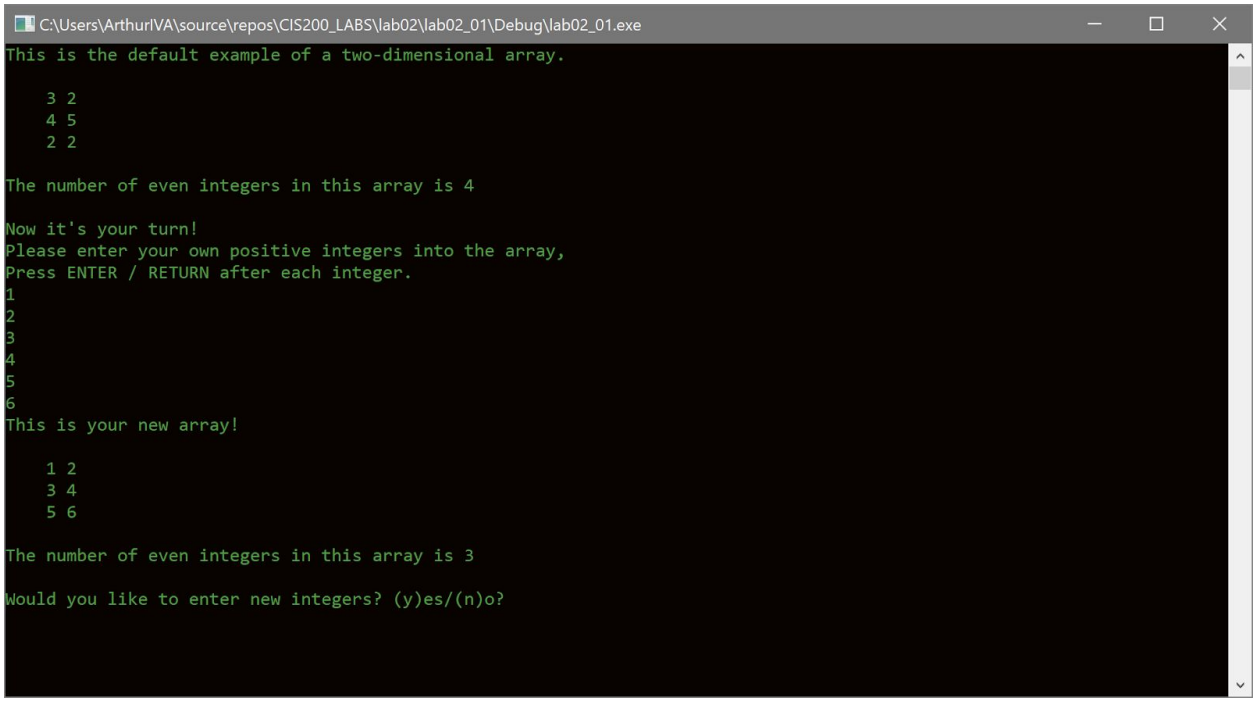
```
//Program Name:
//Programmer Name: Arthur Aigeltinger IV
//Description:
//Date Created: 09/25/18

#include <cctype>
#include <iostream>
#include <iomanip>

#define MAX_ROWS 3
#define MAX_COULMNS 2

//Function Prototypes
int howManyEven(int[][MAX_COULMNS]);
void printArray(int[][MAX_COULMNS]);


int main()
{
	//Initialize default array contents and
	int arrayIntValues[MAX_ROWS][MAX_COULMNS] = { {3,2}, {4,5}, {2,2} };
	int userIn = 0; //NO LONGER NECESSARY
	char userChoice = ' ';

	std::cout << "This is the default example of a two-dimensional array." <<
std::endl;
	printArray(arrayIntValues);

	std::cout << "The number of even integers in this array is " <<
howManyEven(arrayIntValues) << std::endl << std::endl;

	std::cout << "Now it's your turn!" << std::endl;

	//Begin menu for user input
	do
	{
		userChoice = ' ';

		std::cout << "Please enter your own positive integers into the array,"
<< std::endl;
		std::cout << "Press ENTER / RETURN after each integer." << std::endl;

		//Outer loop iterates through rows with formatting.
		for (int i = 0; i < MAX_ROWS; i++)
```

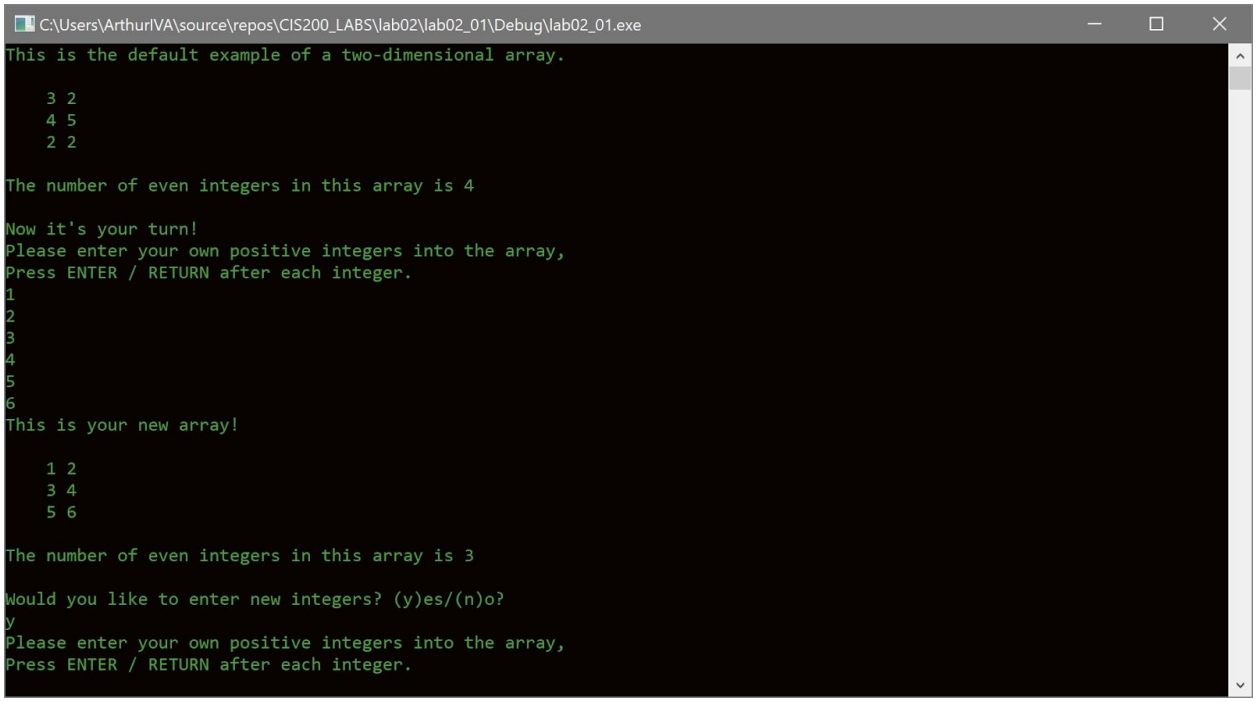
```
		{
			//Inner Loop iterates through sub-columns of array.
			for (int j = 0; j < MAX_COULMNS; j++)
			{
				do
				{
					userIn = 0;
					std::cin >> userIn;

					if (userIn <= 0)
					{
						std::cout << "That was negative... try again
you silly goose!" << std::endl;
					}
				} while (userIn <= 0);

				arrayIntValues[i][j] = userIn;
			}
		}

		std::cout << "This is your new array!" << std::endl;
		printArray(arrayIntValues);
		std::cout << "The number of even integers in this array is " <<
howManyEven(arrayIntValues) << std::endl << std::endl;

		//Choose to continue or not
		do
		{
			std::cout << "Would you like to enter new integers? (y)es/(n)o?"
<< std::endl;
			std::cin >> userChoice;
			userChoice = std::tolower(userChoice);
		} while (!(userChoice == 'y' || userChoice == 'n'));

	} while (userChoice != 'n');

	return 0;
}


//Description: Identifies even integers within a 2 dimensional array and counts how
many exist.
//Pre-Condition: Array that contains integers that can be evaluated as even or odd.
//Post-Condition: Gives nunber of even integers in the given.
int howManyEven(int arrayIntValues[][MAX_COULMNS])
{
	int numEvens = 0;
```

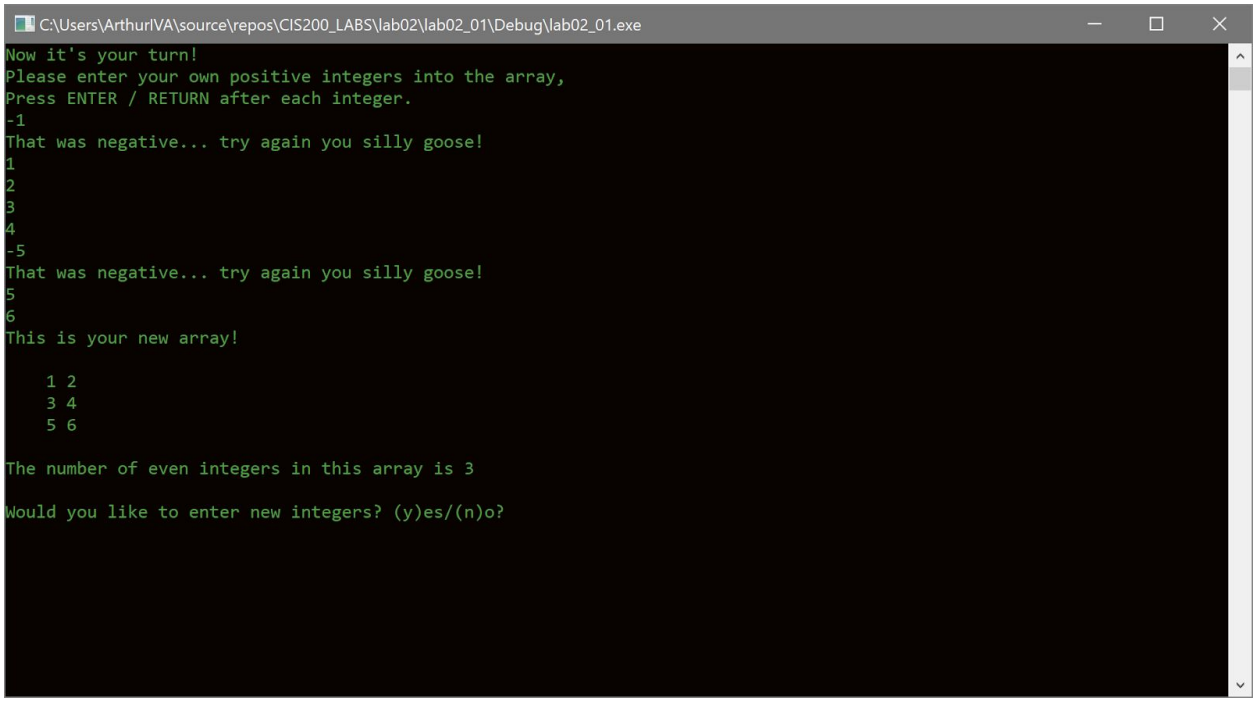
```
	//Outer loop iterates through rows with formatting.
	for (int i = 0; i < MAX_ROWS; i++)
	{
		//Inner Loop iterates through sub-columns of array.
		for (int j = 0; j < MAX_COULMNS; j++)
		{
			if ((arrayIntValues[i][j] % 2) == 0)
			{
				numEvens++;
			}
		}
	}
	return numEvens;
}

//Description: Iterates through two dimensional array and prints it in a formatted
state.
//Pre-Condition: Have filled array that can be printed.
//Post-Condition: Formatted array printed for the user.
void printArray(int arrayIntValues[][MAX_COULMNS])
{
	std::cout << std::endl;
	//Outer loop iterates through rows with formatting.
	for (int i = 0; i < MAX_ROWS; i++)
	{
		//Formatting
		std::cout << std::setw(4) << std::left;

		//Inner Loop iterates through sub-columns of array.
		for (int j = 0; j < MAX_COULMNS; j++)
		{
			std::cout << " " << arrayIntValues[i][j];
		}
		//Drop to next row.
		std::cout << std::endl;
	}
	std::cout << std::endl;
}
```

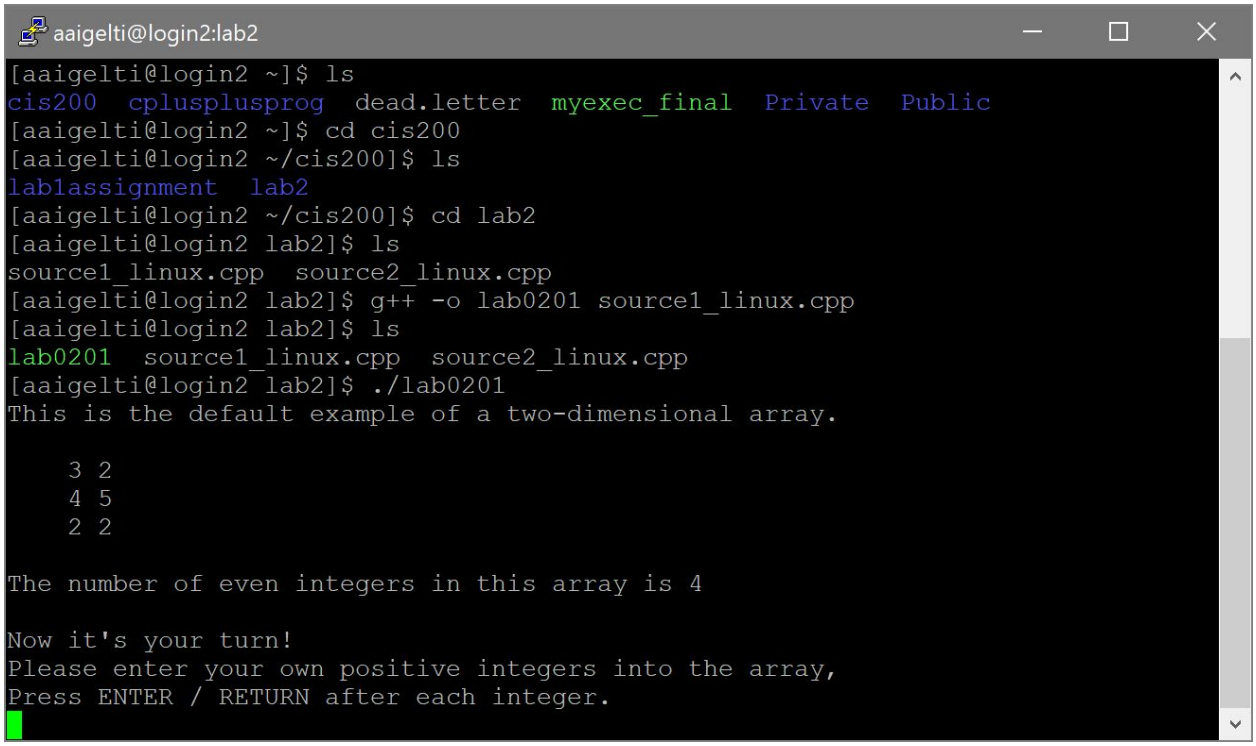
11. Screenshots

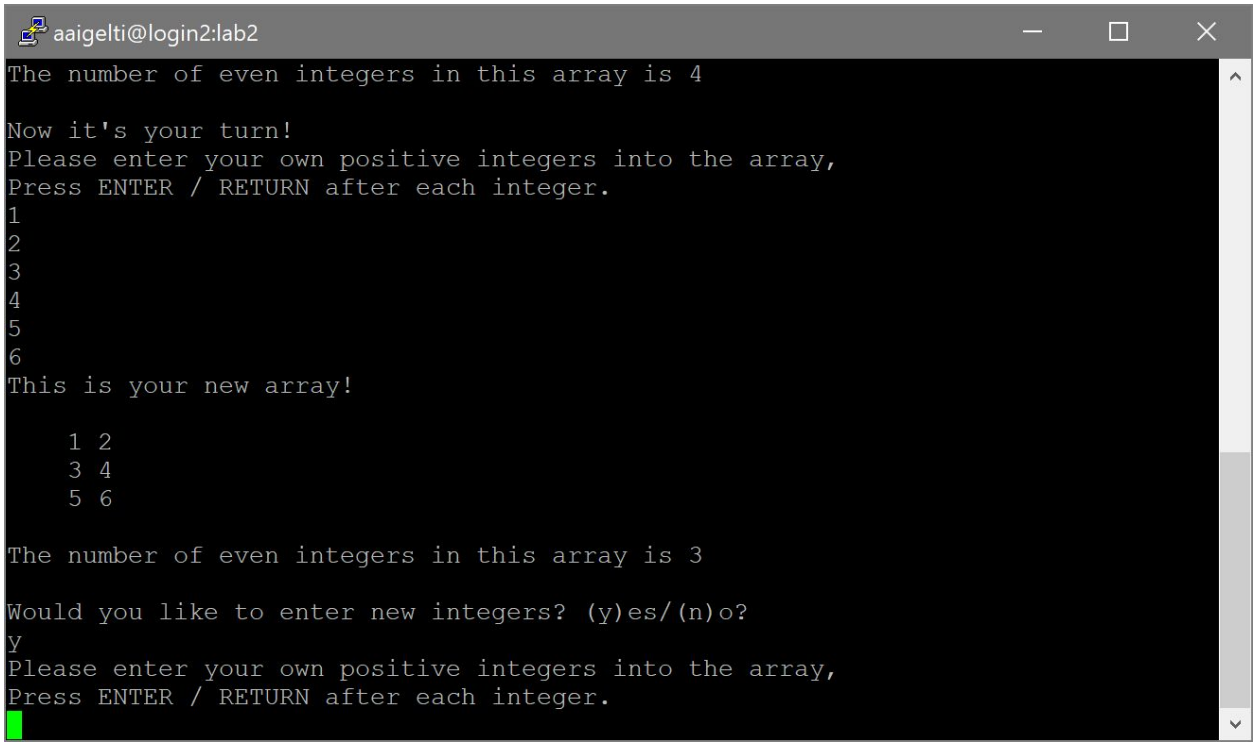


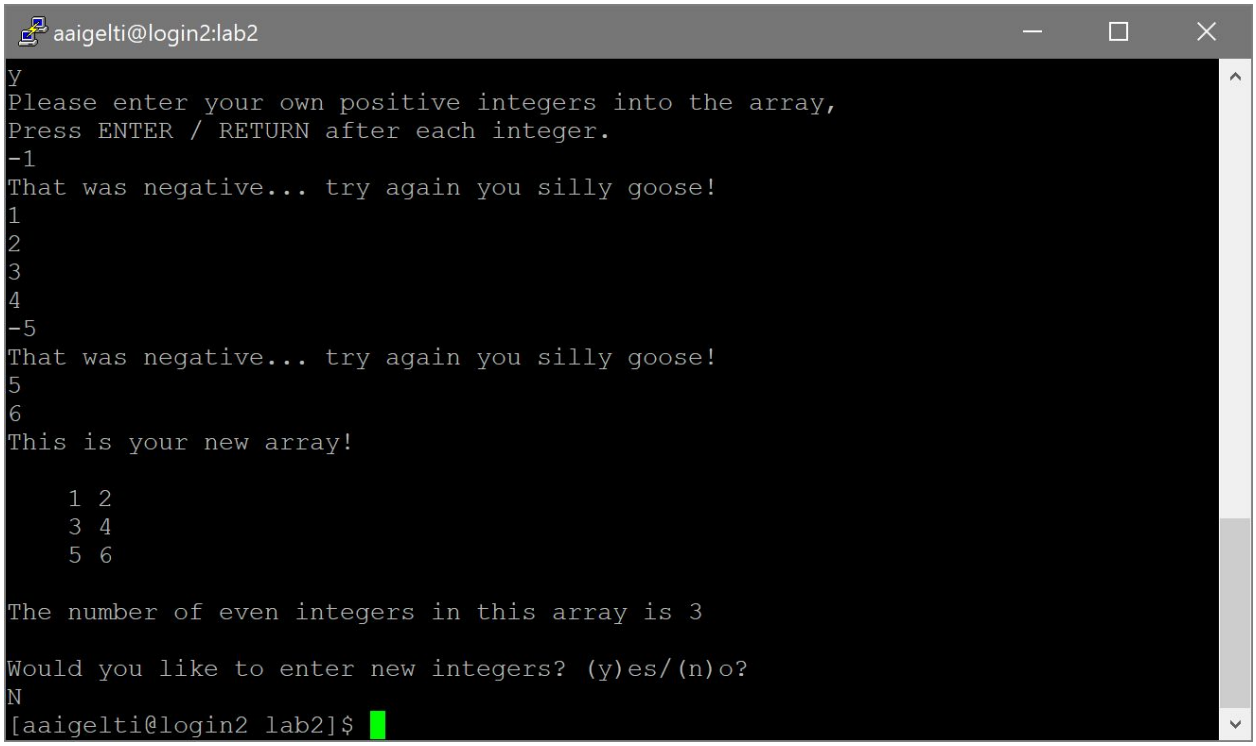



C:\Users\ArthurlVA\source\repos\CIS200_LABS\lab02\lab02_01\Debug\lab02_01.exe
This is the default example of a two-dimensional array.
3 2
4 5
2 2
The number of even integers in this array is 4
Now it's your turn!
Please enter your own positive integers into the array,
Press ENTER / RETURN after each integer.
1
2
3
4
5
6
This is your new array!
1 2
3 4
5 6
The number of even integers in this array is 3
Would you like to enter new integers? (y)es/(n)o?
y
Please enter your own positive integers into the array,
Press ENTER / RETURN after each integer.


C:\Users\ArthurlVA\source\repos\CIS200_LABS\lab02\lab02_01\Debug\lab02_01.exe
Now it's your turn!
Please enter your own positive integers into the array,
Press ENTER / RETURN after each integer.
-1
That was negative... try again you silly goose!
1
2
3
4
-5
That was negative... try again you silly goose!
5
6
This is your new array!
1 2
3 4
5 6
The number of even integers in this array is 3
Would you like to enter new integers? (y)es/(n)o?


aaigelti@login2:lab2
[aaigelti@login2 ~]$ ls
cis200 cplusplusprog dead.letter myexec_final Private Public
[aaigelti@login2 ~]$ cd cis200
[aaigelti@login2 ~/cis200]$ ls
lab1assignment lab2
[aaigelti@login2 ~/cis200]$ cd lab2
[aaigelti@login2 lab2]$ ls
source1_linux.cpp source2_linux.cpp
[aaigelti@login2 lab2]$ g++ -o lab0201 source1_linux.cpp
[aaigelti@login2 lab2]$ ls
lab0201 source1_linux.cpp source2_linux.cpp
[aaigelti@login2 lab2]$ ./lab0201
This is the default example of a two-dimensional array.
3 2
4 5
2 2
The number of even integers in this array is 4
Now it's your turn!
Please enter your own positive integers into the array,
Press ENTER / RETURN after each integer.


aaigelti@login2:lab2
The number of even integers in this array is 4
Now it's your turn!
Please enter your own positive integers into the array,
Press ENTER / RETURN after each integer.
1
2
3
4
5
6
This is your new array!
1 2
3 4
5 6
The number of even integers in this array is 3
Would you like to enter new integers? (y)es/(n)o?
y
Please enter your own positive integers into the array,
Press ENTER / RETURN after each integer.



13. Status

Program is operational in both Windows and Unix. Plus I still have all of my hair, so that's good.