**Lab 1 Report**

**Name : Shashwat Mishra**

**UC email : mishras8@mail.uc.edu**

1. Objectives and concepts explored in this assignment included debugging a file ( especially in task 2 and 3), using structs, and getting familiar with Visual Studio IDE. Using the Visual Studio IDE, I was able to use various tools and features such as debugging, finding errors easily as it shows which line of code has errors. Debugging the file and learning how to set breakpoints in Task1 understand how we can debug a very big file by setting breakpoints which basically stop / break the execution of code on a particular line. Using structs I was able to reduce to complexity of code and make it more readable.

These concepts are important in this course and for a career in CS as software engineers deal with a huge code-base with hundreds and thousands of lines of code. It is very difficult to read through the whole code base to find errors. Thus, debugging comes in very handy in these situations. Making the code more readable by using structs is also very helpful because when we collaborate with other developers, it will be easy for them to quickly go through the code.

A screenshot of a computer

Description automatically generated

The figure above shows the compiled code after debugging the file from TASK 1.

1. In Task 2, I first compiled the program and saw that a lot of the fields in the output were either missing values or simply had wrong values. The total sales was zero and the divisions in the output were not updating. As mentioned in the task, I started debugging the file by stepping in and then going through each line until wrong output was displayed in the terminal. I then went ahead and changed the code wherever I found the errors.

The programmer might have made those mistakes due to lack of testing and debugging the code. These mistakes can be avoided by testing the code by using methods such as unit testing or by debugging the file so that such bugs can be eliminated.

A screenshot of a computer

Description automatically generated

The figure above shows the output of TASK 2 after debugging the file and fixing the errors.

1. In task 3, the first step I did was to go over the whole code and see what the output was. I then understood what all variables and arrays were used. Then I followed the instructions given and created a struct called “Product” and an instance called “prod\_array”. The struct “Product” included variables id,units,prices and sales. These variables were arrays in the old program and now I have implemented them as variables under the struct.

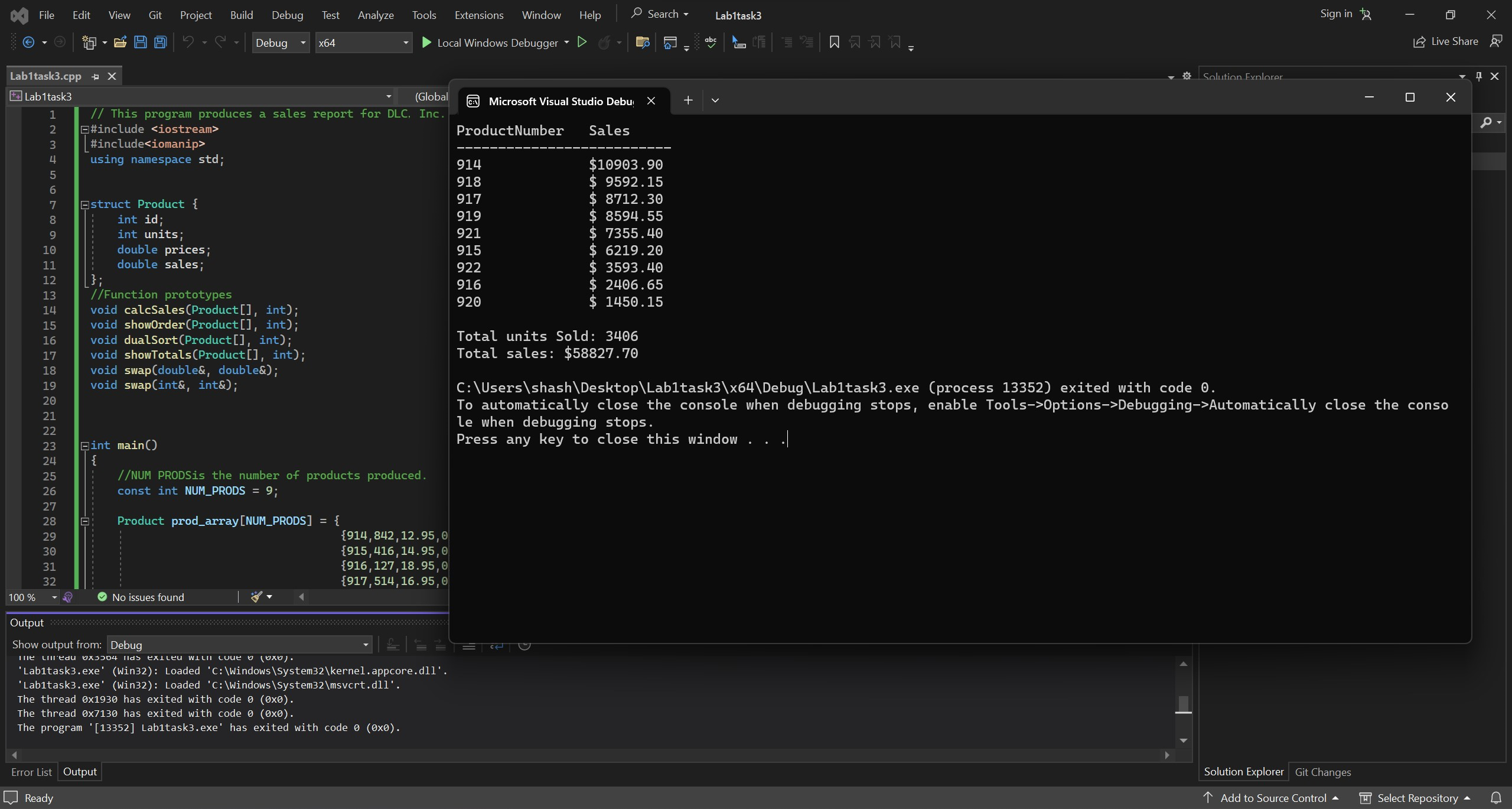
Next step I created an instance of the “Product” struct called the “prod\_array” which is an array with 9 values ( NUM\_PRODS = 9). Then I added all the data into the array from the other 4 arrays and removed the other 4 arrays as all the product information was saved in prod\_array.

Next, I changed the functions arguments and variables.

id[index] was changed to prod\_array[index].id, sales[index] was changed to prod\_array[index].sales and so on…

These changes made sure that all the data was accessed through the prod\_array and not the other 4 arrays.

In the end, I ran into few errors and bugs such as not changing the arguments of the functions correctly and when declaring each function using wrong datatypes instead of Product struct. These bugs and errors were fixed by debugging the code line by line until the code could compile successfully and it produced the expected output.



The figure above shows the output of TASK 3 after changing and modifying the code by using structs.