CALCULUS PROJECT (MT-224) …

Contents

[Objective of the Problem: - 1](#_Toc74605312)

[Analytical Solution: 2](#_Toc74605313)

[Instruction Manual and Example: 4](#_Toc74605314)

[Explanation of the Matlab Commands used: 4](#_Toc74605315)

[Detailed Example of program: 5](#_Toc74605316)

[Flow Chart: 7](#_Toc74605317)

[Conclusion: 8](#_Toc74605318)

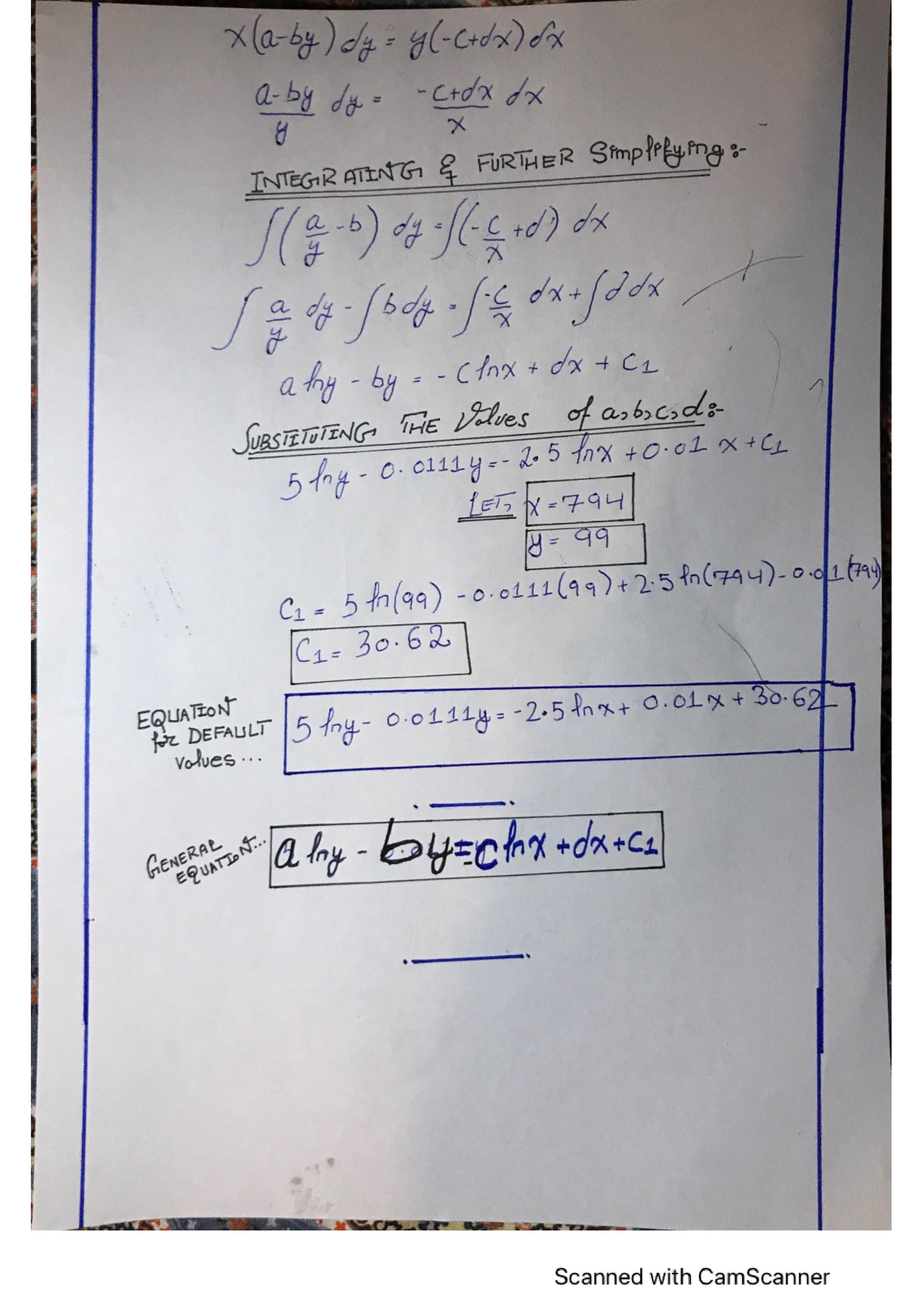
[Contributions Section: 8](#_Toc74605319)

[Difficulties and how they were overcome: 8](#_Toc74605320)

# Objective of the Problem: -

Our client is the owner of an aquarium and he needed to hold a single endangered species. But now as the number of fishes increased in the aquarium so the owner introduced another species which will act as a predator for the original species (fishes) so our client demands software from our software company that maintains balance between two species so none of them goes extinct, for that purpose it is desired to make a model that clearly illustrates the population of both species over the period of five years. So, we make a program that either uses default values or asks the user to enter the initial/ boundary values and display the desired result (population).

# Analytical Solution:



# Instruction Manual and Example:

1. When our program runs it displays a welcome message in a box and besides that it displays our Company Name and group member names as well.
2. Then you have to Press “Ok” key to continue and then press any key when prompt “Welcome to the Program! Press any key to continue” is shown on the screen.
3. Then you have to enter your choice that if you want to use default values or enter and set the initial conditions. You have to enter “1” for adding input or enter “0” for using the default values.
4. When you enter “1” it asks for input and you have to enter “number of prey” and “number of predators”.
5. When you enter “0” it uses default values already set and display the graphs.
6. After entering the values program will show you the required graphs showing predator/ prey population over time.
7. After displaying these graphs program asks for your option that if you want to restart the program again or you want to exit.
8. In case you press restart again it starts right from the very first point and displays name of your company and group members again.
9. If you press exit program ends by showing the thank you message “Thank you for using our program”.

# Explanation of the Matlab Commands used:

1. **Input (prompt):**

This displays the text enclosed within brackets and asks the user to provide input. For example: input (hello) displayed ‘Press any key to continue’ and waited until user pressed a key for input.

1. **While-end loop:**

While-end loop is used when looping is needed, and it runs until the given expression is proven false. It is used in situations where it is not known how many times the program will loop over as it is user controlled. The loop continues until a stated condition is being satisfied.

1. **Plot ():**

This was used to plot 2-D line plot of matrices of equal size. Additional line properties were also specified.

1. **X-label and Y-label:**

These were used to label the axes of our plot of graph and the values of the labels change with the values entered by user.

1. **Title ():**

This was used to give our second graph a title.

1. **Ui-wait:**

To not allow program to run until user has pressed ok on a message box.

1. **Msgbox:**

It creates a separate window to display any message to the user.

1. **Inputdlg:**

It creates a separate window to take input from the user.

1. **Str2double:**

It converts a string into a double.

1. **If-end:**

To write code that only runs on a conditional basis.

1. **Function-end:**

It is to create a function so that we can write code and call it in multiple places.

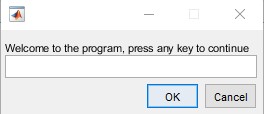
1. **Switch: -**

It is to choose between different options or cases when we are having many conditional cases.

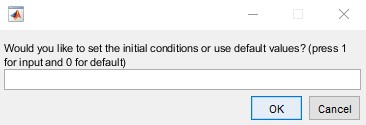
# Detailed Example of program:



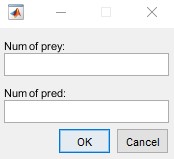
A Dialogue box that opens up what the program is initially run for the first time.

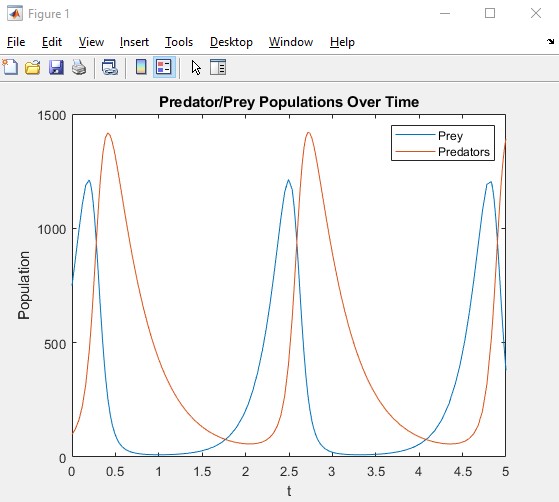
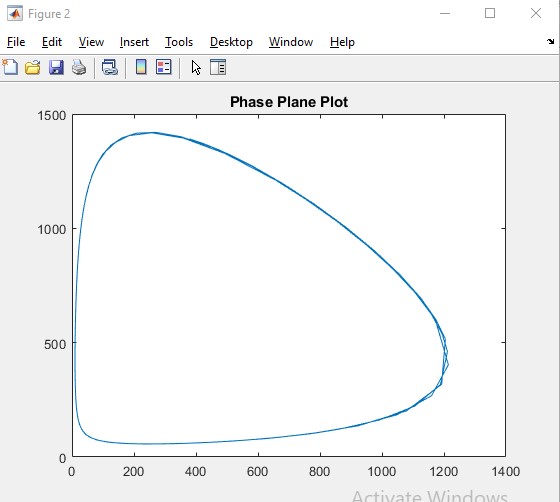


Another dialogue box that prompts the user to enter any key to continue.

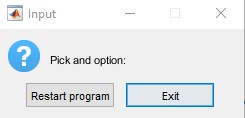


A third dialogue box asking for the user whether they wish to use the default values or to set their own initial conditions.

 The following dialogue box that shows up if the user decides to enter 1 in the previous dialogue box.



After the user enters their values, the DE equation is solved using ODE-45 and the program plots the Lotka Volterra line and phase graph as shown above. This is what the dialogue box then shows.



Finally, this is the last dialogue box that shows that: it asks the user whether they want to restart the program or to terminate it.

# Flow Chart:

# Conclusion:

After writing our solutions we found out that the Lotka Volterra equation seems to not always hold up for large values of population unless we change our constants. We feel that it might not be the most accurate method to map population of a predator-prey dynamic as it does not account for environmental and human factors. Hence we feel that for small values of population in a controlled environment the Lotka Volterra equation can prove to be very useful and accurate, however, when dealing with larger populations that have more factors to take into account, we suggest that the research team uses a different method or model to map the relationship between the prey and its predator.

# Contributions Section:

1. **Usman Ali Bokhari**
   * Wrote Matlab Code for the given problem statement.
   * Contributed in finding right values of constant to display proper graphs.
   * Contributed in pointing out and correcting some errors in the equation.
2. **Sherwan Qadir**
   * Contributed in finding and writing neat analytical solution.
   * Contributed in writing objective, proper instruction manual and functions of commands in the program.
   * Edited some Matlab code to terminate the program and display boxes.
3. **Shamaiem Shahid**

* Contributed in making neat code flow diagram using Word.
* Compiled report on MS Word.
* Contributed in writing analysis of the problem.
* Edited Matlab code to restart or terminate using while loop conditions.

# Difficulties and how they were overcome:

* First of all, one of our group member used Mathematica in last semester but was having some knowledge of Matlab through the assignments being done in this semester. So, for this we learned new Matlab graph commands and also taught those commands to him.
* We were facing some difficulties because graph was being displayed in wrong format so for this we modified code and added proper values in code and used ode-23 to get our desired graph.
* Also a difficulty was found in implementing the termination condition so we applied a correct condition in the program that made it working perfect.
* We also found difficulty in labeling the graphs with changing values so we tried many commands such as x and y label commands, text commands etc. and finally we got our desired output and correct labeling of graphs.