## Executive Summary

Learning how to derive functions is a task that most students struggle with. Derivatives are a step away from what most students have seen in their math classes so far, and because they're not solved strictly arithmetically at an undergraduate level, they not only provide new concepts that students must grasp, but they are also not solvable through a normal calculator. There are many websites that teach students the rules for differentiation, but only few that will calculate them. There are no website that allow students to enter the equations to derive into a calculator in the same format as they are written by hand, or in a textbook. That compounds the problem. Additionally, while there are websites that are dedicated to derivatives, there are few options to find practice problems with answers for students wanting to practice on their own in preparation for a test.

In order to solve this and help students actually learn the material, I implemented a four piece solution into my project that I feel more than adequately addresses and resolves the problems students face while learning derivatives, as well provides a template that can be used in a variety of academic settings. The four pieces are divided onto four tabs as seen below:
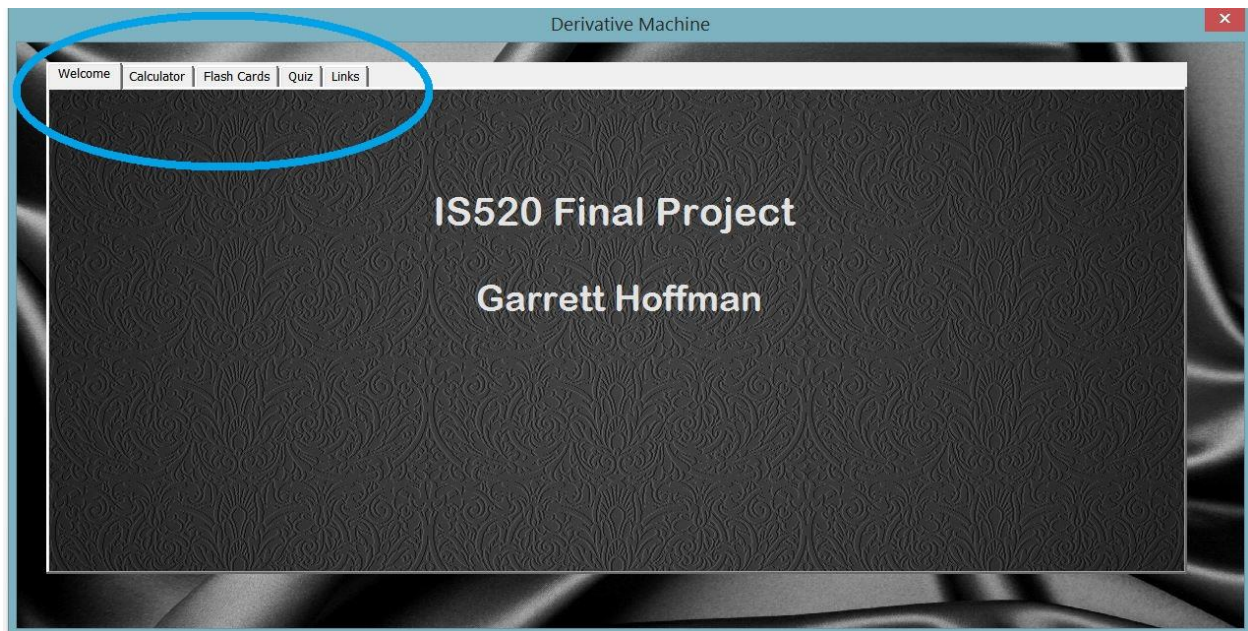
The four piece solution that I created encompasses the following parts, and each is explained in detail in the Implementation Documentation section of the write-up.

1. Derivative Calculator

2. Customizable Flash Cards

3. Customizable Quizzes

4. Links to continue learning

Implementation documentation

Derivative Calculator

A major problem that students face while learning derivatives is coming up with the correct solution in the first place. In this scenario, it's very difficult to learn them. Since derivatives are unique, they cannot be solved with a normal calculator, and must be derived either by hand, or on a few select websites online. In order to solve this problem, I created a calculator using VBA that allows students to enter a derivative problem into the program, and the VBA code takes the input that was entered and embeds it into a website's URL, and pulls the answer from the web. One of the main advantages of designing the calculator in this way is that 1st through 10th order derivatives can be found, and as derivatives are not strictly arithmetically solved, sometimes finding answers can be difficult, but with the calculations pulled from a solid website, the answers are always correct. I also designed the calculator so that the input is displayed in such a way that students enter the derivatives in the same way that they appear in textbooks or on paper. They can use this calculator to help them solve full or partial problems, and can even use it while also utilizing other features of the program, such as the flash cards or quiz. To use the calculator, the student clicks on the 'Calculator' tab at the top of the program. Then they enter the derivative that they want solved into the space provided. In order to make the input look like textbook problems do, the fields are designed so that they must click on each area of the input box (which is really multiple boxes placed next to each other to appear as one), and enter the appropriate values into each section of the input area. I designed it with the exponent in

the upper right corner where it would really be, rather than make them use a caret, since that notation is not used in writing or in textbooks. I also hard coded parenthesis and the X variable in the proper location as well. Once they have entered the derivative that they want solved, they can select which order of derivative they would like, from first to tenth, with first being the default. The user then presses the 'Calculate' button, and the VBA code takes the input entered by the user and embeds it into a URL and retrieves the answer from the internet and displays it to the user. I tried to make the format and presentation of the calculator look as professional as possible.
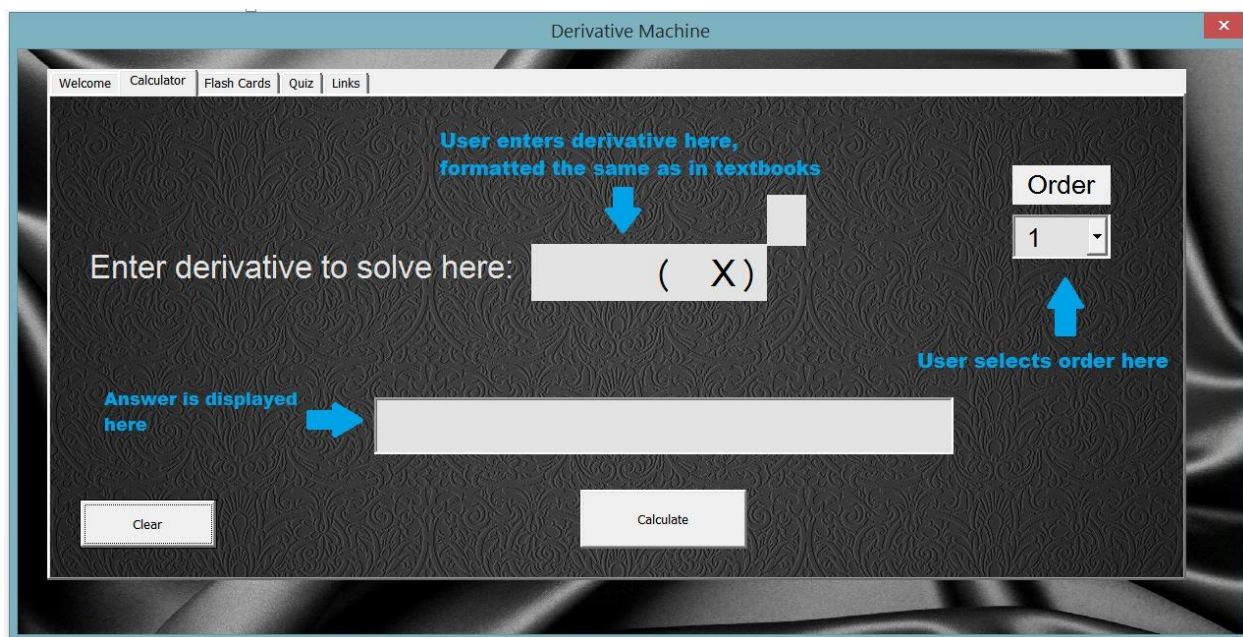
Customizable Flash Cards

In order to effectively learn a new concept in math, practice is absolutely necessary. Recognizing this fact, I created customizable flash cards using VBA. In order to make customizing cards as easy and intuitive for the user as possible, I pull the questions and answers for the cards from a spreadsheet labeled 'Flash Cards'. There is a column labeled 'Questions', where the user enters the questions that should appear on the front of the cards, and there is another column labeled 'Answers', where the user inputs the answers that will appear on the back side of the cards. That is all done in the in the backend, before the VBA code is run. When the user runs the VBA code, the program starts and the user can click on the flashcard tab to practice with flash cards. The user then needs to press the 'Start' button, which activates the 'Flash Cards' sheet in Excel, and moves the active cell to A2, where the first question is. The user then works out the derivative problem on their own, and when they think they have the correct answer, they click the 'Flip Card' button to see the answer and check theirs. This then moves the active cell over to the answers column, and sets the textbox that makes up the card to show the contents of that cell. I wrote the code for the 'Flip Card' button to never change rows so that the no matter how many times the button is pressed. I used an IF statement to check which column the active cell is in, and if it is the Questions column, the code will tell the active cell to move to the Answers column, and if it is already in the Answers column, it will do the opposite. While designing this part of the project, I kept in mind that although this project is set up for derivatives right now, it could easily be used

for another, non-math related subject, so I made the flash cards are easily customizable as possible.
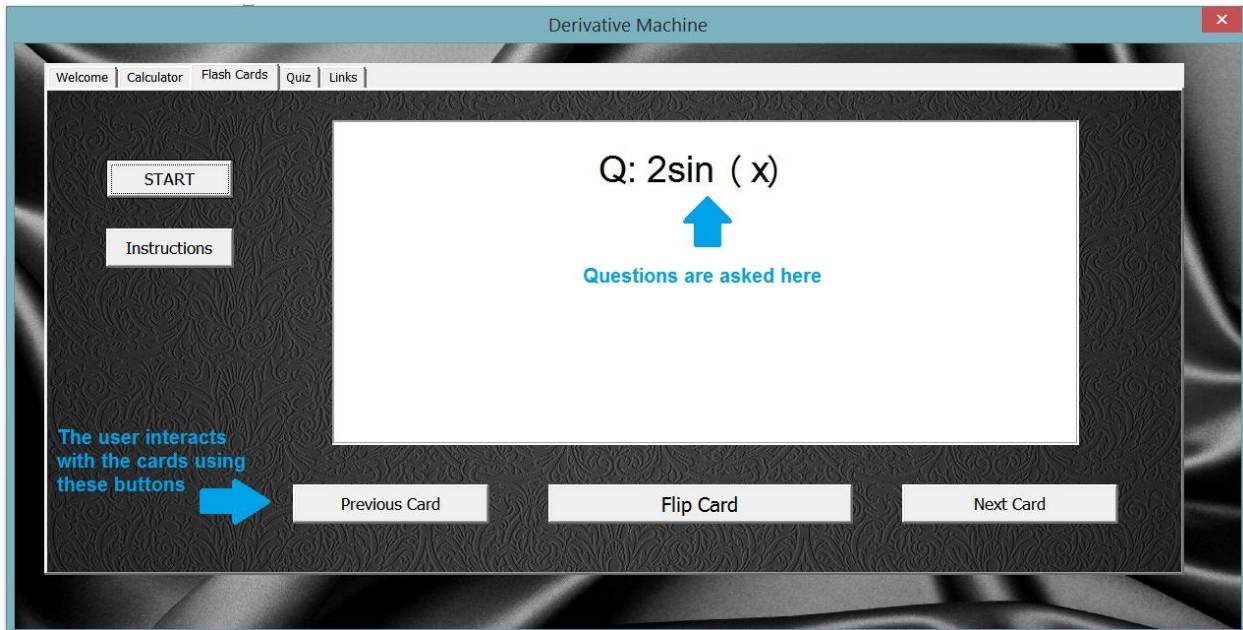


```
Private Sub btnFlipCard_Click()
If Selection.Column = 1 Then
  ActiveCell.Offset(0, 1).Select
    With UserForm1
      lblCardContent = Sheets("FlashCards").Range("B" & ActiveCell.Row).Text
    End With
Else
  ActiveCell.Offset(0, -1).Select
    With UserForm1
      lblCardContent = Sheets("FlashCards").Range("A" & ActiveCell.Row).Text
    End With
```

This code snipet alternates what the card displays, the question or the answer, and never changes the active cell's row, so the proper answer will always be with the corresponding question

Customizable Quizzes

In order to more fully learn the material that the student is studying, I created a quizzer as part of the project. It allows the user to create customized quizzes. It takes the questions and the answers that the user enters, displays the question and provides the user with a textbox in which to enter the answer. When the user has entered an answer to the quiz question, they press the button 'Submit Answer'. The VBA then takes the answer that the user entered, and compares it with the customizable answer key. The VBA then displays a message back to the user, which informs them if the answer they inputted was correct, or incorrect. If the user does not know the answer to the quiz question, then they can press the 'Show Answer' button to see what it is so that they can keep learning. To do this, I wrote code that finds the answer to the corresponding question, and displays it in the answer input box.
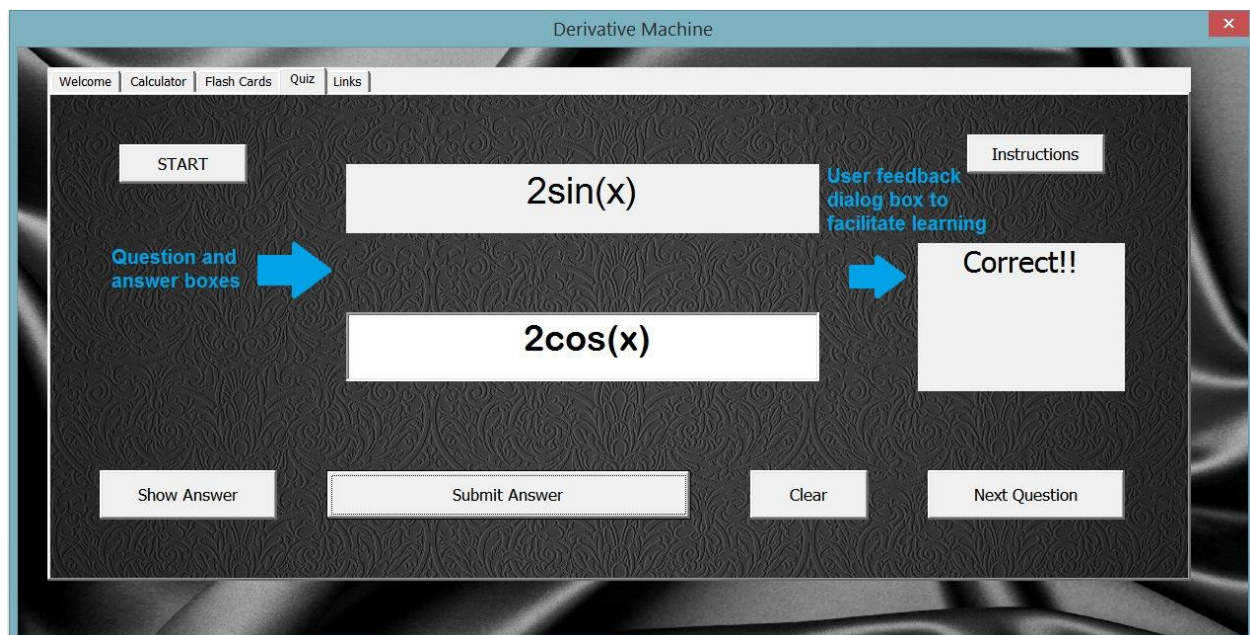
```vba
Private Sub btnSubmitAnswer_Click()

Sheets("Questions").Range("D" & ActiveCell.Row).Value = txtboxUserAnswer.Value

If Sheets("Questions").Range("c" & ActiveCell.Row).Value = Sheets("Questions").Range("D" & ActiveCell.Row).Value Then

    With UserForm1

        lblCorrectIncorrect = "Correct!!"

    End With

Else

    With UserForm1

        lblCorrectIncorrect = "Not quite, Please try again"

    End With

    Exit Sub

End If
```

This code snipet compares the answer that the user entered with the answer from the answer key. If the answers are the same, the user told that it is correct. If they are not the same, the user is also alerted to that.

Links to Continue Learning

In order to help whichever student that is using this program continue learning and have a way to find additional information when they get stuck, I put in a tab that can contain either instructions or links. This page is updated dynamically through the VBA code. There is a sheets titled 'Links' that the user has access to, and the content on this sheet is tied to the content on the Links spreadsheet page. In this example, I put some links to websites that provide additional help and instructions on derivatives, but the content on this tab can be dynamically changed to anything the user wants to display. I added this functionality to make the program more user friendly.
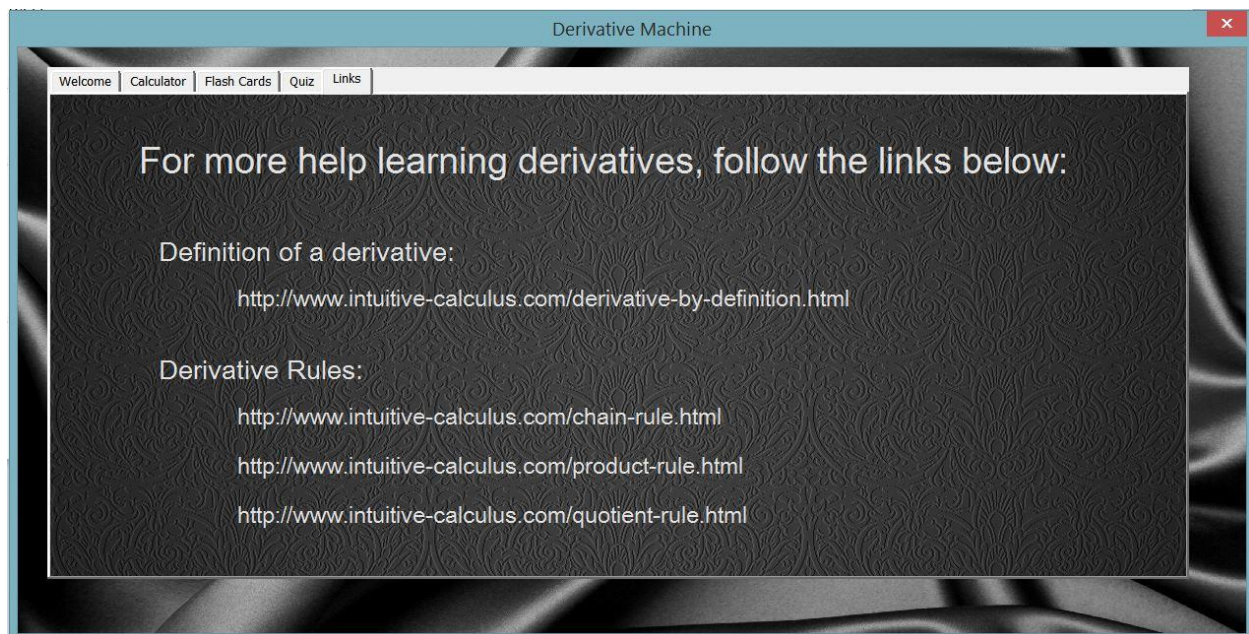

```
'Set the links (or other content on the Links page at startup
lblTitle1.Caption = Sheets("Links").Range("A2").Text
lblTitle2.Caption = Sheets("Links").Range("A3").Text
lblLink1.Caption = Sheets("Links").Range("B2").Text
LblLink2.Caption = Sheets("Links").Range("B3").Text
lblLink3.Caption = Sheets("Links").Range("B4").Text
lblLink4.Caption = Sheets("Links").Range("B5").Text
Application.Visible = False
```

This code snipet shows how the links are created dynamically based on what the user wants them to be and has inputed into the spreadsheet. This is in the form's initialization sub procedure so that they appear at startup

Learning and Conceptual Difficulties Encountered

This was the first project I've ever done in VBA where I decided what to do, and what features to implement. I did all of the projects that were assigned for homework, but there were always specific instructions for them. I learned a lot about VBA, VBA code, and what you can and can't do with VBA while doing this project. There are many things that I found out that VBA can do that I didn't know before. I also learned how involved and how much time even a small project can take to accomplish. It takes a lot of time to write the actual VBA code, and it also takes a significant amount of time to research how write the code that you want the program to do. For example, my project starts by clicking a button on one of the spreadsheets. At first, I just wanted excel to not go to any other sheets while the user form was running, so that the user couldn't see any of the answers to the flash cards or quizzes since they are stored on spreadsheets in the workbook to make it easy for the user to customize them. However, as I got more involved in the project, I decided that I actually wanted Excel to close, and for the user to only be able to see the user form while it was running, but then for Excel to show up again when the user form was closed. I knew that Dr. Allen had done something similar to that when we were working with Internet Explorer, so I started researching how to make that happen. I was able to find the answer eventually, but it took time than I was expecting a question like that to take. However, I learned that it is possible to do, and I learned that most questions that I have about VBA can be answered by searching online for a solution. I also learned a lot about user forms while doing this assignment. I had only used user forms once before, when we were doing the user form project. However, that project was only about connecting content from the user form to a

spreadsheet, and from a spreadsheet to the user form. I went much deeper in my project and I incorporated tabs, picture backgrounds, etc. I also had to learn how to many other small things that involve user forms such as dynamically changing the text on labels. It is done differently than changing the text of textboxes. I also learned how integrated a spreadsheet can be with user forms, and how nice it is to have content in the user form stored on a spreadsheet and called, rather than hard coding anything into the form. I also learned more about puling data from the web, and what types of URLs make it easier to do if the sites that you are pulling content from have flash or JavaScript running on them. For example, when I was building the calculator portion of the project, the first site that I tried to use to find the answers to derivatives was wolframalpha.com. However, since the URL doesn't change on that site based on the derivative question that it is solving, it could not be used, because it wouldn't allow the user of my project to input any derivative problem and solve it. So, I had to find a website that has dynamically changing URLs depending on what derivative problem is being solved. That allowed me to dynamically create a separate URL for every derivative calculation that a user requests an answer for. I also learned the value of keeping my code clean. For small projects that only have a few sub procedures in them, it isn't as big of a deal, but when the code starts to get long, it's very nice when all of it is formatted nicely.

Assistance

I did not receive any help from anyone else while doing this project. I thought of the idea, and I referenced my old projects and the internet when I needed help. I was able to figure out an answer to every question that I had by either referencing a project that we have already done this semester, or by searching the internet for an answer.