I SYS 520 FINAL PROJECT:

AUTOMATING BENJAMIN GRAHAM'S 10 HURDLE CHECK FOR STOCKS

By: Philip Jones

Executive Summary

Background

In today's financial world, prospective investors constantly seek ways to quickly and efficiently obtain information regarding the quality of various investment opportunities. For years, the stock market has been one of the most common sources of such investing. Often considered the father of modern financial analysis, Benjamin Graham devised a list of ten "hurdles," or benchmarks, by which one could use to measure the overall quality of a stock to determine whether it would be a good investment. Inspired by the class's Fallen Angel assignment, I created this project to test these hurdles when given a valid ticker symbol for a publicly traded company.

For reference, a list of Benjamin Graham's Ten Hurdles is included below:

1. An earnings-to-price yield at least twice the AAA bond rate.
2. P/E ratio less than 40% of the highest P/E ratio the stock had over the past 5 years.
3. Dividend yield of at least 2/3 the AAA bond yield.
4. Stock price below 2/3 of tangible book value per share.
5. Stock price below 2/3 of Net Current Asset Value.
6. Total debt less than book value.
7. Current ratio greater than 2.
8. Total debt less than 2 times Net Current Asset Value.
9. Earnings growth of prior 10 years at least at a 7% annual compound rate.
10. Stability of growth of earnings in that no more than 2 declines of 5% or more in year-end earnings in the prior ten years are permissible.

Solution Overview

This project includes using a series of three web queries to effectively gather the data as well as a user form to (1) allow the user to enter a valid stock ticker symbol, (2) execute the web queries, and (3) display the results of the analysis. The project essentially consisted of the following five parts: Part 1 automated a web query for a particular stock provided by the user; Part 2 automated a query that retrieved the latest AAA bond information; Part 3 consisted of a query that retrieved a company's most recent balance sheet line items; Part 4 analyzed the data from the three queries and used the results to determine whether the stock passed the different hurdles; and Part 5 consisted of loading that information into the user form and creating custom buttons on the ribbon for the user's convenience.

Implementation Documentation

PART 1: WEB QUERY FOR STOCKS

Analogous to the Fallen Angel assignment, I generated a web query for a particular stock from www.reuters.com (Yahoo Finance would have worked equally well, but Reuters contained a bit more information and was more efficient to use) and customized the query so that the stock ticker symbol for the company of interest is included in the appropriate spot of the URL and therefore automatically retrieves the latest information for a particular company.
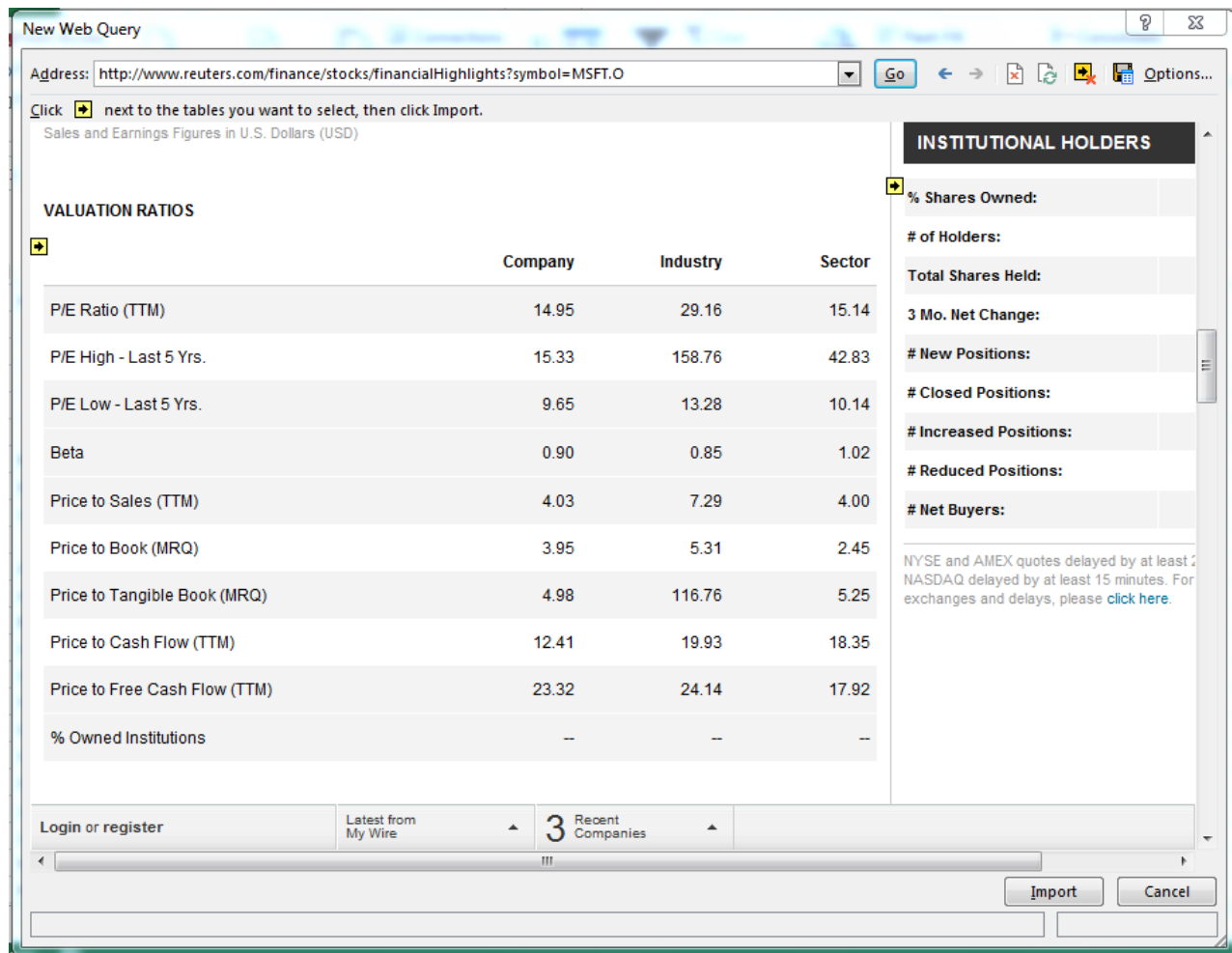
Figure 1.1: A quick glance at the initial Reuters web query. I imported more data than what is shown in the picture.

```vba
Sub getQueryData()

    Sheets("DataCollection").Select
    Sheets("DataCollection").Range("B1").Select
    Selection.Copy

    Sheets("WebQuery").Select
    Range("A1").Select
    ActiveSheet.Paste
    Application.CutCopyMode = False
    Range("A6").Select


With Selection.QueryTable
        .Connection = _
    "URL;http://www.reuters.com/finance/stocks/financialHighlights?symbol=" & Range("A1").Value
    .WebSelectionType = xlEntirePage
    .WebFormatting = xlWebFormattingNone
    .WebPreFormattedTextToColumns = True
    .WebConsecutiveDelimitersAsOne = True
    .WebSingleBlockTextImport = False
    .WebDisableDateRecognition = False
    .WebDisableRedirections = False
    .Refresh BackgroundQuery:=False
End With
```

Figure 1.2: Initial code for the first web query.

PART 2: WEB QUERY FOR AAA BONDS

I then created a second web query on another worksheet that imported the most recent

data for AAA-bonds from the St. Louis Federal Reserve website (http://research.stlouisfed.org/

fred2/series/DAAA).

```
Sheets("BondQuery").Select
Range("A6").Select
With Selection.QueryTable
        .Connection = _
    "URL;http://research.stlouisfed.org/fred2/series/DAAA"
    .WebSelectionType = xlEntirePage
    .WebFormatting = xlWebFormattingNone
    .WebPreFormattedTextToColumns = True
    .WebConsecutiveDelimitersAsOne = True
    .WebSingleBlockTextImport = False
    .WebDisableDateRecognition = False
    .WebDisableRedirections = False
    .Refresh BackgroundQuery:=False
End With
```

Figure 1.3: Code for the second web query.

PART 3: WEB QUERY FOR A COMPANY'S BALANCE SHEET INFORMATION

Similar to the first query, I programmed the macro to use a company's stock ticker in the

web URL to obtain the latest information regarding a company's balance sheet line items. These

data were retrieved from Yahoo Finance. Screenshots of the code and web query are not

included because they are almost identical to those in Part 1.

PART 4: EXTRACTING DATA FROM WEB QUERIES FOR HURDLE TESTS

This part of the project was by far the most difficult, tedious, and time consuming. Once the three web queries were generated and the latest data were available, I then programmed Excel to find the specific values of interest and then copied and pasted them all onto one worksheet. The following table includes a list of all the entries that were required for calculating the hurdles.

Stock Price	P/E Ratio	E/P Ratio	AAA Bond Yield	5-Year High P/E	Total Assets	Total Liabilities
Total Debt	Dividend Yield	Growth Rate	Current Ratio	Short-Term Debt	Long-Term Debt	Total Shares Held
Net Tangible Assets	Current Assets	Current Liabilities	Book Value	Net Current Asset Value	Cash and Cash Equivalents	Short-Term Investments
Net Receivables	Inventory	Other Current Accounts	Accounts Payable	Short/Current Long-Term Debt	Other Current Liabilities	

Table 1.1: A list of entries required for calculating the hurdles.

To accurately pull in these data into their respective fields on the "Data Collection" worksheet, I repeatedly used several "find" commands that located a unique cell containing the name of the entry (e.g., stock price), offset the proper number of cells (determined by working with the web query worksheets), and then selected, copied, and pasted the respective values. An example of such a process is included below in Figure 1.4 where the P/E ratio is extracted from the web query. The process for the other 26 entries in Table 1.1 is analogous.

```
Cells.Find(What:="P/E Ratio (TTM)", After:=ActiveCell, LookIn:=xlFormulas _
    , LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
    MatchCase:=False, SearchFormat:=False).Activate
ActiveCell.Offset(0, 1).Select

Selection.Copy
    Sheets("DataCollection").Select
    ActiveCell.Offset(2, 0).Select
    ActiveSheet.Paste
```

Figure 1.4: Code for locating the P/E ratio in the web query, copying the value, and pasting it into the proper cell in the "Data Collection" tab.

After several hours of extracting all of the necessary values, I then completed all

other calculations that were necessary for evaluating the ten hurdles. For instance, many

stocks' balance sheets do not include a total value for current assets, though they include totals

for all the line items that comprise total assets. Given this information, I automated the

spreadsheet to sum up these constituent parts to obtain the current assets. Once I had these, I

formatted a list of cells to indicate "Yes" or "No" depending on whether the stock passes

particular hurdles. Figure 1.5 shows a portion of the table of extracted/calculated values and

the table of whether a stock passed the hurdles in the Data Collection tab. For sake of

convenience in the user form (see next section), I included code to copy-paste-value the results

of passing the hurdle (i.e., "Yes" or "No") into a list below that would be used to populate the

user form. Doing this pasted the values "Yes" or "No" without the formulas.

	A	B	C	D	E	F	G	H	I
1	Stock Ticker Symbol:	HOG							
2	Stock Price	68.71							
3	P/E Ratio	21.02					Hurdle #	Hurdle Description	Pass Hurdle?
4	Earnings-to-Price	0.047573739					1	Earnings-to-Price >= 2*(AAA Bond Yield)	No
5	AAA Bond Yield	4.3					2	Fallen Angel? (P/E Ratio >= 0.4*(5-Year High P/E))	No
6	5-Year High P/E	83.32					3	Dividend Yield >= (2/3)*(AAA Bond Yield)	No
7	Fallen Angel?	Yes					4	Stock Price < (2/3)*(Net Tangible Assets)/(Total Shares Held)	No
8	Total Assets	9,405,040					5	Stock Price < (2/3)*(Net Current Asset Value)	No
9	Total Liabilities	6,395,554					6	Total Debt < Book Value	No
10	Total Debt	5,259,170					7	Current Ratio > 2	No
11	Dividend Yield	1.6					8	Total Debt < 2*(Net Current Asset Value)	No
12	Growth Rate	14.54					9	Growth Rate >= 7	Yes
13	Current Ratio	1.59					10	Data are unavailable to test.	Unknown
14	Short-Term Debt	1,842,457							
15	Long-Term Debt	3,416,713					Hurdle #	Pass Hurdle?	
16	Total Shares Held	186,113,903					1	No	
17	Net Tangible Assets	2,979,034					2	No	
18	Current Assets	3,988,803					3	No	
19	Current Liabilities	2,509,586					4	No	
20	Book Value	3,009,486					5	No	
21	Net Current Asset Value	-1,270,367					6	No	
22	Cash And Cash Equivalents	1,211,419					7	No	
23	Short Term Investments	99,009					8	No	
24	Net Receivables	2,138,376					9	Yes	
25	Inventory	424,507					10	Unknown	
26	Other Current Assets	115,492							
27	Accounts Payable	667,129						Total Number of Cleared Hurdles	1

| | TenPoints | **DataCollection** | WebQuery | BondQuery | BalanceQuery | ⊕ | | ◄ | |

READY

Figure 1.5: A screenshot of the Data Collection tab that lists the extracted/calculated values on the left and the stock hurdle results on the right.

PART 5: LOADING THE USER FORM AND CREATING CUSTOM TAB & BUTTONS

Once the hurdles were calculated, I created a user form and then included code that would load the form with the corresponding "Yes" or "No" strings for the hurdles. I also showed both the number of calculable hurdles and the number of hurdles passed for each stock, as seen in Figure 1.6 below.

Figure 1.6: This is what the completed user form looks like. The results shown are for Microsoft (MSFT). Passing 6 hurdles is considered to be very strong relative to other publicly-traded companies. In contrast, other companies like Harley-Davidson (HOG) consistently pass only 1 hurdle and are therefore deemed poor performers.

To get the appropriate responses into the user form above, I used the following code:

```
lblHurdle1 = Sheets("DataCollection").Range("H16").Value
lblHurdle2 = Sheets("DataCollection").Range("H17").Value
lblHurdle3 = Sheets("DataCollection").Range("H18").Value
lblHurdle4 = Sheets("DataCollection").Range("H19").Value
lblHurdle5 = Sheets("DataCollection").Range("H20").Value
lblHurdle6 = Sheets("DataCollection").Range("H21").Value
lblHurdle7 = Sheets("DataCollection").Range("H22").Value
lblHurdle8 = Sheets("DataCollection").Range("H23").Value
lblHurdle9 = Sheets("DataCollection").Range("H24").Value
lblHurdle10 = Sheets("DataCollection").Range("H25").Value

lblTotalCount = Sheets("DataCollection").Range("I27").Value & Sheets("DataCollection").Range("J28").Value & _
Sheets("DataCollection").Range("I29").Value
'
```

Figure 1.7: Code for populating the user form labels with the appropriate responses.

In addition to populating the user form, I also wanted to create a custom tab in Excel that would contain two different buttons: (1) An "Instructions" button that would produce a message box containing adequate instructions for running the program, and (2) a "Get Stock Info" button that would bring up the user form and allow the user to input his/her desired stock ticker. To do this, I used the Custom UI editor and used the XML code as shown in Figure 1.8.
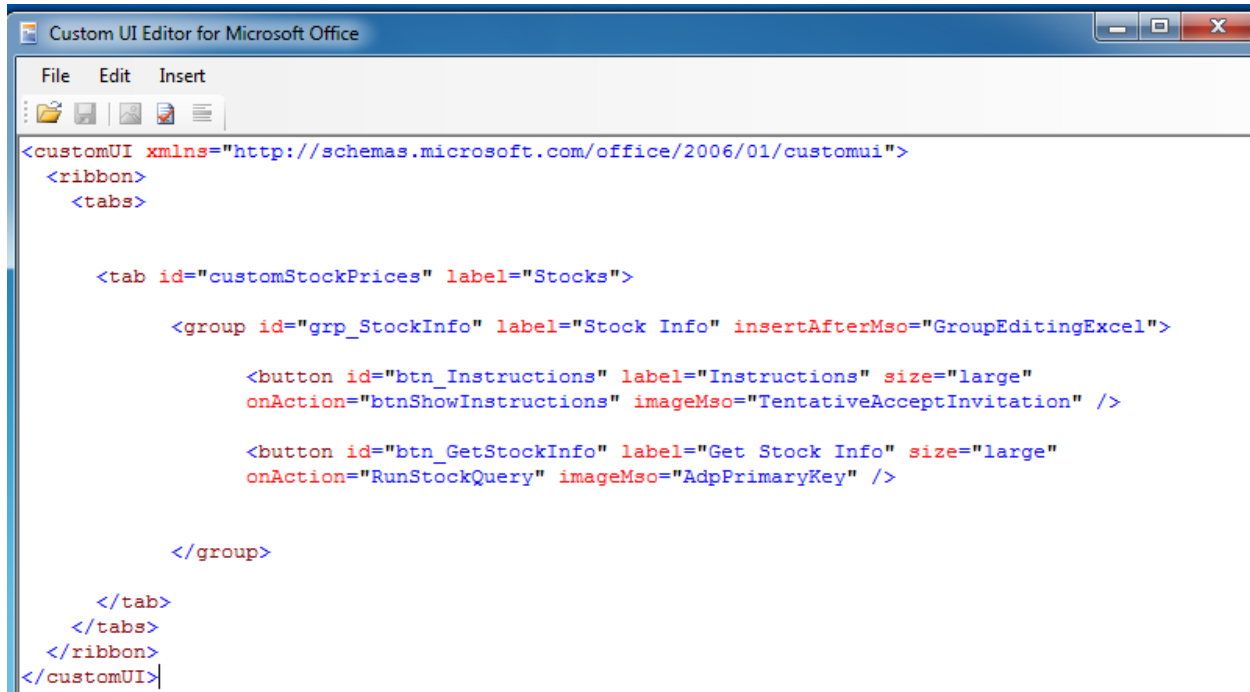


```xml
<customUI xmlns="http://schemas.microsoft.com/office/2006/01/customui">
  <ribbon>
    <tabs>

      <tab id="customStockPrices" label="Stocks">

        <group id="grp_StockInfo" label="Stock Info" insertAfterMso="GroupEditingExcel">

          <button id="btn_Instructions" label="Instructions" size="large"
          onAction="btnShowInstructions" imageMso="TentativeAcceptInvitation" />

          <button id="btn_GetStockInfo" label="Get Stock Info" size="large"
          onAction="RunStockQuery" imageMso="AdpPrimaryKey" />


        </group>

      </tab>
    </tabs>
  </ribbon>
</customUI>
```

Figure 1.8: XML code used in the Custom UI editor used to produce custom tab and buttons.

Doing this, I was able to create a "Stocks" tab with two buttons, as shown below in
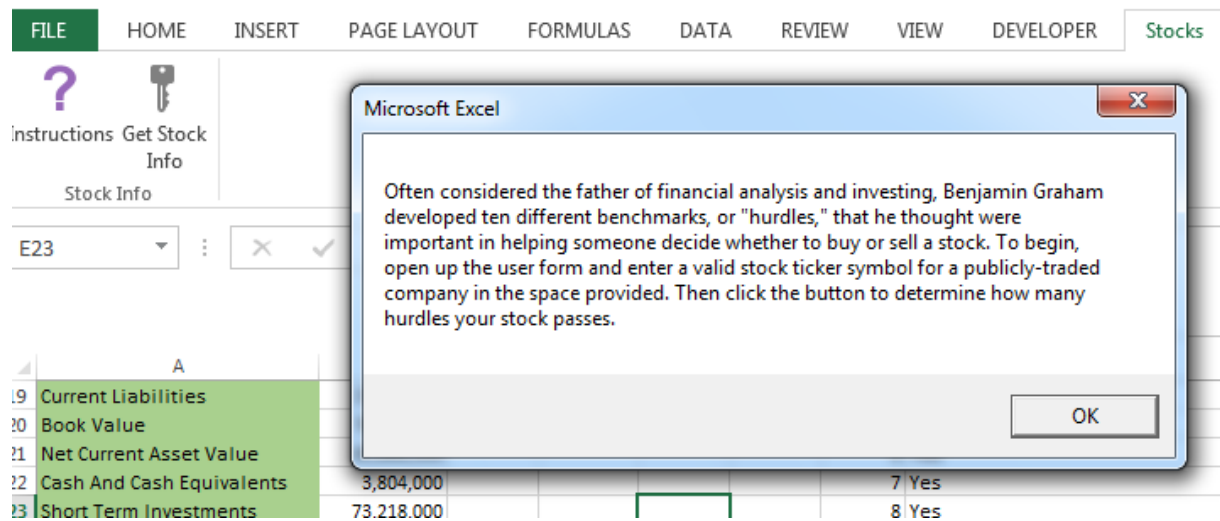
Figure 1.9.

Figure 1.9: A view of the Stocks tab (upper right) with the "Instructions" and "Get Stock Info" buttons (upper left). When the user clicks the instructions button, the message box above appears. When the user clicks the "Get Stock Info" button, the user form (shown in Figure 1.6) appears.

Discussion of Learning and Conceptual Difficulties Encountered

As aforementioned, the purpose of this project was to produce a useful, automated tool that would provide useful information regarding a publicly traded company's performance in the stock market for potential investors. Most of the initial obstacles that I encountered were overcome by gaining a better understanding of each of the ten hurdle requirements and how to calculate them. As a result, I learned a great deal about these finance and accounting concepts.

Unfortunately, the tenth hurdle proved to be much too difficult to test. This hurdle requires one to find the year-end earnings for a company and compare it to past earnings to determine whether it had dropped more than 5 percent at any time within the past ten years. My complications arose not from actually calculating the data, but from not being able to find a

website that contained the data that could be used in a web query; the data just simply aren't available in a way in which VBA automation can be used. Because of this, I noted that the data were unavailable and listed the result of the tenth hurdle as "unknown." Thus, the maximum number of calculable hurdles was always nine instead of ten.

Web Queries

One of the biggest difficulties in scraping the data from the web queries is that both Reuters and Yahoo Finance do not always have the same data for companies in the same format. For instance, in the balance sheet web query via Yahoo Finance, some companies have their current assets and current liabilities listed explicitly. Other companies, however, such as JPMorgan Chase & Co. (JPM) had a "—" listed in their current asset value and instead listed values for the individual components that comprise their current assets. As a result of this inconsistency, I simply chose to have the macro sum up these constituent parts and place this sum as the current assets amount regardless of whether the amount was explicitly stated or not.

Hurdle Calculation

As shown on page 2, the third hurdle states that a company's dividend rate must be greater than 2/3 of the AAA bond yield. However, after considerable research on this matter, I was unable to find a specific answer as to whether companies that don't pay dividends (e.g., Google) automatically fail this hurdle or whether they should simply not test this hurdle. Because I could not find a clear answer one way or the other, I took the former approach and declared that a non-dividend paying company does not pass the third hurdle. However, I could

easily change this in the future through a simple if-then statement that tells Excel to place an "unknown" in the result window for a stock that does not have a dividend yield. Doing so would show the total number of passed hurdles as being out of 8 instead of 9 for such companies. While this situation does not imply an error with my VBA code, the decision of what to in the case of non-dividend paying companies was a theoretical difficulty specific to the financial field.

Assistance and Acknowledgements

To complete this portion, I first consulted with Professor Gove Allen on project details to effectively obtain the proper scope for the project. To get ideas for how I would format my program, I viewed the project details of Joshua Lindsay, a former BYU student who completed this project in 2012 for this class. Though I viewed his code to gather general ideas, I did not copy his code and simply use it as mine; instead, I looked for inconsistencies and inefficiencies in his code and sought to create a more efficient program that performed the same duties with less lines of code. For instance, instead of using some of the sites that he required multiple web queries for a particular value of interest, I used the Reuters page because it contained more information on the same page, which allowed me to have less "Web Query" worksheets in my workbook. Although, like Mr. Lindsay, I was unable to calculate the tenth hurdle, I am confident that my program is cleaner, more efficient, and represents my own honest, individual efforts.

As mentioned above, I initially had to spend a good amount of time studying the components of the ten hurdles and understand what I needed to calculate. For some of the

balance sheet items, I consulted with a close friend currently in the Junior Core of BYU's

Accounting program and with various online sources.