VBA for Finance Engineering

Overview

What this Course Is

This is a VBA course with applications in financial engineering. It does not require any previous programming knowledge. Basic college level mathematics are required. The goal of this course is to instill a practical understanding of VBA in the context of how it is used in the financial industry. Students can expect to come out with the ability to use VBA for their first day on the job as well as impress interviewers with their VBA knowledge.

What this Course Is Not

This is not an end-to-end academically-oriented VBA course. It does not contain any Computer Science theory and will not give a deep understanding of the mechanics of VBA outside of traditional finance useage.

Prerequisites

College level mathematics

Knowledge of basic financial instruments

Syllabus

LEVEL 1: EXCEL FUNCTIONS

1.1: INTRODUCTION

1.2: WORKSHEET FUNCTIONS

1.3: MATH/LOOKUP FUNCTIONS

1.4: COMPOUNDING

1.5: INTEREST RATES

1.6: DISCOUNT FACTORS

1.7: BOND PRICING

1.8: OFFSET

1.9: PIVOT TABLES

1.10: HOMEWORK PREP

LEVEL 2: VBA BASICS

2.1: SUBS/FUNCTIONS

2.2: VARIABLES

2.3: VECTORS

2.4: LOOPS

2.5: BOND PRICING

2.6: STATEMENTS

2.7: LINEAR INTERPOLATION

2.8: LINEAR INTERPOLATION - CODE

2.9: FORWARD RATE CURVE

2.10: FORWARD RATE CURVE – CODE

2.11: HOMEWORK PREP

LEVEL 3: OPTION PRICING

3.1: OPTION PAYOFF MODEL

3.2: BINOMIAL ASSET PRICING MODEL

3.3: BINOMIAL ASSET PRICING MODEL - CODE

3.4: BLACK SCHOLES

3.5: HOMEWORK PREP

LEVEL 4: CLASSES

LEVEL 5: DATA STRUCTURING

How to Progress

- Watch all videos for a single chapter. You should proceed in the provided chapter order.
- Try out the examples demonstrated in the videos
  1. Play around with the examples; make changes, etc., to get an intuitive feel for things.
  2. The lecture example code is deliberately not provided (except in rare cases), as it is important to code things up manually to gain a better understanding.
- Ask questions to the instructor or in the comments section of a video until everything is clear.
- Do the HW for the level. Feel free to ask questions on the video or to the instructor
- Repeat for the next level.

How to Ask Questions

- Questions may be on the lectures, material, or exercises.
- Search the related video to see if your question has already been asked/answered.
- If it has already been answered, feel free to ask any follow-up questions to clarify if still unclear.
- If it has not already been answered, post a new question on the video or to your instructor.
- Posts related to exercises should not contain any code; they should be conceptual in nature and/or descriptions (in English) of issues encountered. Instructor responses will also be conceptual in nature – they will guide you to finding the solution on your own, without providing the solution. Posts that do not conform to this standard will be removed.
- Students are encouraged to respond to other students’ queries, publicly and within these guidelines.

Recommended Study Planner

Total time for the course is 8 weeks. After 8 weeks, you will lose instructor support, the ability to earn the certificate, and access to the course material.

You may work at your own pace within the allotted 8 weeks, but the below is the recommended pace per level:

Week 1: Chapter 1

Week 2: Chapter 2

Week 3: Chapter 3

Week 4: Chapter 4

Week 5: Chapter 5

Week 6: Case Study

Week 7: Case Study

Week 8: Case Study (yes, it might take 3 weeks)

Submission Instructions

- Submissions should be for an entire chapter at a time (no individual sections or exercises).
- Every exercise should be submitted as its own script(s).
- You must wait for feedback on your submissions prior to advancing to the next chapter, as it will be locked until the instructor unlocks the next chapter

Grading Policy

- Once you submit your exercises for a given level, your instructor will provide detailed feedback on your code. This will usually occur within 24-48 hours of submission; often quicker, rarely slower.
- Each exercise is assigned a grade out of 100. The score for a chapter is the straight average of all the exercises in that chapter.
- Each chapter (and the final project) is assigned an overall course weight. Your final score is the weighted average of all the chapters and final project. The chapter weights are equal.
- You will need a minimum of a 70 weighted average score to earn the certificate. A minimum score of 90 will earn you a certificate with distinction.

Grading Criteria

- Each exercise is graded out of 100. An exercise that is code correct and meets the specification of the instructions gets an 80. The extra 20 points are devoted to code commenting, code format, code conciseness/clarity, efficiency of code, and taking the optimal coding approach (within what was already taught).
- You are expected to build upon your knowledge as you progress through the course. Therefore, once a concept is taught, you are expected to incorporate it into your future code when applicable. Taking an earlier (non-optimal) approach will cost you points.