IE523: Financial Computing
Fall, 2019
Programming Assignment 9: Pricing European- and American-Options Using the Trinomial Model with Memoization
Due Date: 15 November 2019

©Prof. R.S. Sreenivas

In the previous programming assignment you used Dynamic Programming to compute the price of an American-Option to overcome the inefficiency of Recursive methods that do the same. You were able to comfortably run large number of Divisions, to get at an accurate estimate of the price.

In this assignment, you are going to make the Recursive methods efficient by using *Memoization*. You have done this already in the Programming Assignment that computed the price-of-admission to a stylized Card-Game. You are going to do the same – only this time, it is for Recursive procedures that compute the price of Options.

Write a C++ program that computes the price of an

1. European Option, and

2. American Option,

which runs on a command-line, and takes the following variables as input (maintain this order; otherwise, you will loose points!)

1. Expiration,

2. #Stages,

3. Risk-free rate (Continuous compounding) ,

4. Volatility (expressed as a fraction of initial price of underlying),

5. Initial Price of underlying, and

6. Strike Price.

using the Trinomial Model. It is imperative that your code uses memoization, so that we can run large number (≈ 1000) of stages.

The European Option Price has be compared with the Black-Scholes Formula (cf. figure 1). For the American Option Price, I am looking for something like what is shown in figure 2. You can ignore the *put-call parity* verification, if you wish.

Figure 1: Running-time illustration for pricing an European Option using a 5000 stage (Memoized) Trinomial Model.

Figure 2: Running-time illustration for pricing an American Option using a 5000 stage (Memoized) Trinomial Model.