Algorithms Analysis & Design
Project Proposal

Aditya Harikrish

2020111009

1 Introduction

Dynamic programming is a widely used problem-solving tool that often greatly helps in reducing the effort needed to solve computational problems. It can also be unintuitive to beginners.

2 Proposal

I plan on making a beginner's guide to dynamic programming. To build up on the prerequisites, I will introduce time and space complexities, basic examples of dynamic programming where it outperforms some of the competing algorithms, such as the Fibonacci sequence. I will then formally define dynamic programming, explain how a solution could be intuitively and originally thought of and what kinds of problems can be solved using dynamic programming. Some of the problems / algorithms I wish to cover are:

1. Fibonacci sequence

2. Frog Jump

3. Longest increasing subsequence

4. Longest common subsequence

5. Shortest common supersequence

6. Minimum coins

7. Longest palindromic substring

8. Knapsack

9. Tower of Hanoi (one of my all-time favourite problems)

10. Egg dropping problem

11. Bellman Ford Algorithm

12. Floyd Warshall Algorithm

I shall go through memoisation as well in the process. If possible, I would like to cover Optimal binary search trees.

3   Timeline

The tentative timeline is as follows. I would like to finish most of the theoretical part including the prerequisites, along with 5 or so problems by the first week of November. I plan to finish the project by including more problems and if needed, polishing the theoretical parts by the third or fourth week of November.