DOKUZ EYLUL UNIVERSITY

ENGINEERING FACULTY

DEPARTMENT OF COMPUTER ENGINEERING

CME 2204 ALGORITHM ANALYSIS

Lecturers

Asst. Prof. Zerrin Işık

Arş.Gör. İBRAHİM BERBER

Res.Asst. Ali Cüvitoğlu

ASSIGNMENT-2

Dynamic Programming & Greedy Approach

By
YASSER EL HASAN
2019510006

CONTENTS

01 INTRODUCTION

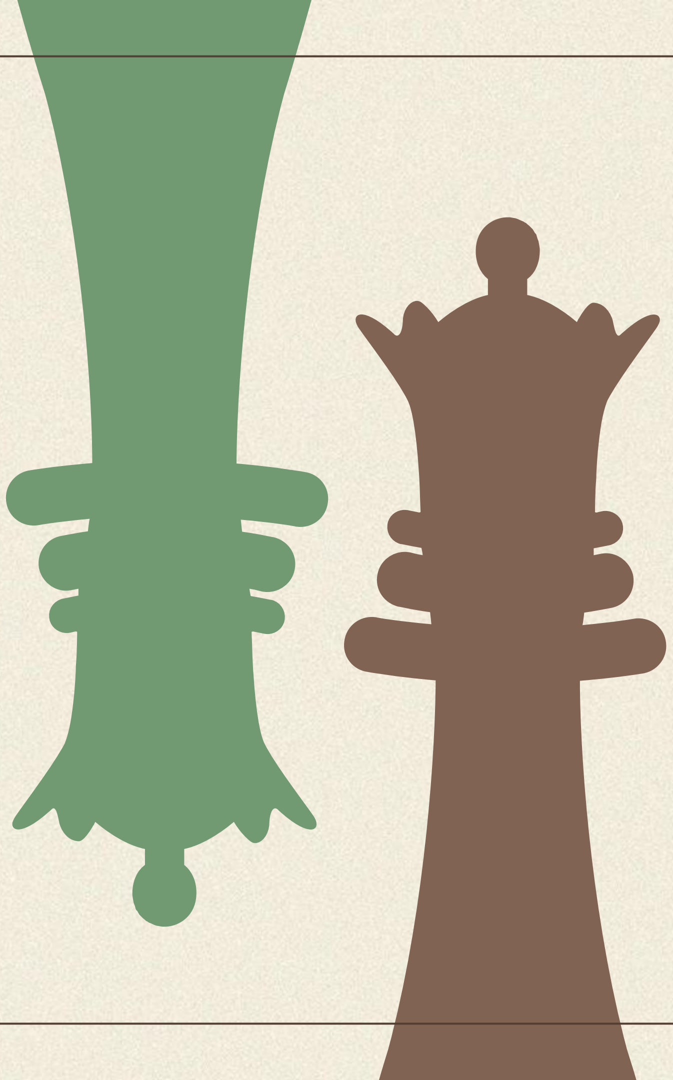
02 DYNAMIC APPROACH

03 GREEDY APPROACH

04 RANDOM APPROACH

05 TIME TABLE

06 CONCLUSION

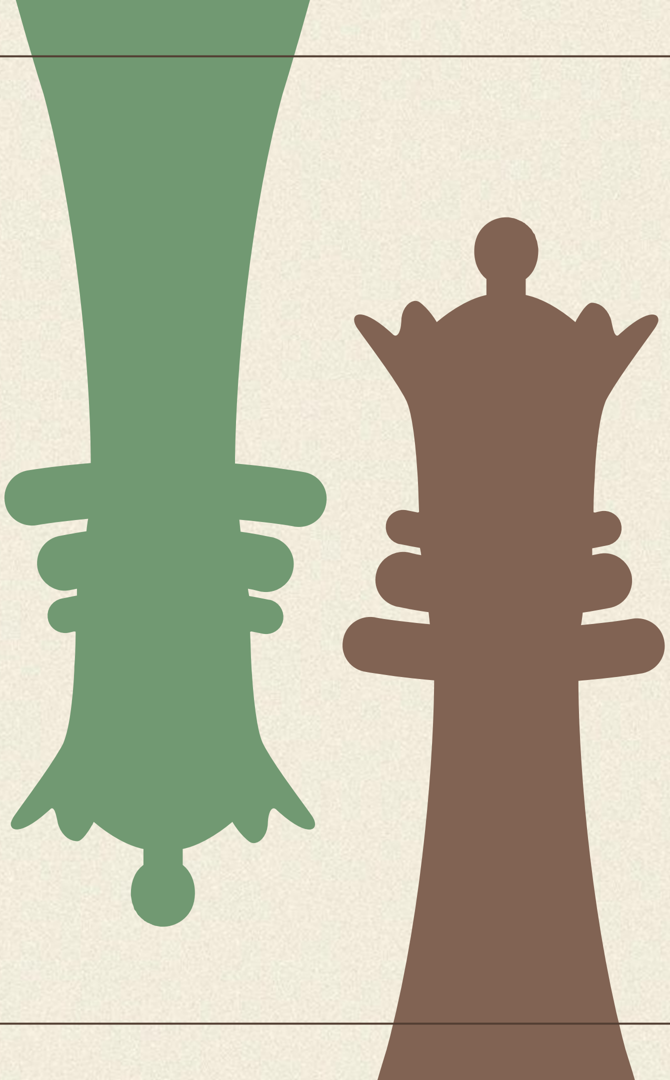


01

INTRODUCTION

INTRODUCTION

In this assignment we will implement a program which simulates a value game using customized chess pieces. In our case, there will be many available pieces for each type of piece, each having a historical name, along with an arbitrary attack point. Each piece type will have a level. You will select your pieces depending on maximum allowed level .The aim of the game is to select pieces with possible attack points.



02

DYNAMIC APPROACH

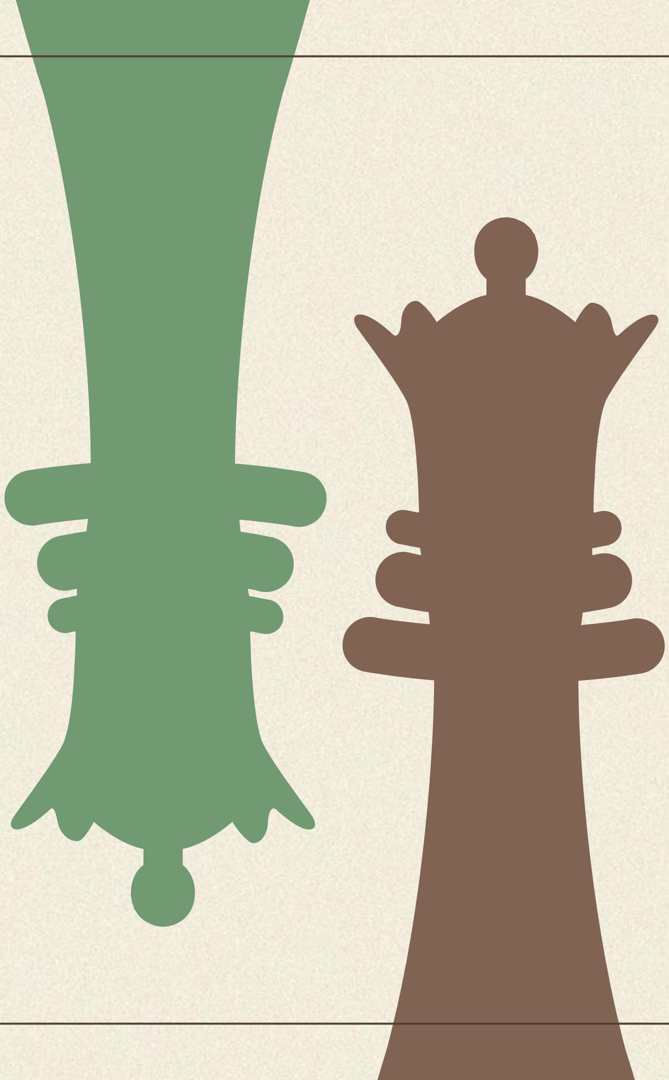
DYNAMIC APPROACH

Dynamic programming approach is similar to divide and conquer in breaking down the problem into smaller and yet smaller possible sub-problems. But unlike, divide and conquer, these sub-problems are not solved independently. Rather, results of these smaller sub-problems are remembered and used for similar or overlapping sub-problems.

So depending on the above definition , I used this approach in the assignment. I create a table that stores the results for subproblems, so I can use them again.

The last result will be concluded in the last cell of the table in the bottom.

A problem that I faced is I could not include only one piece for each type.

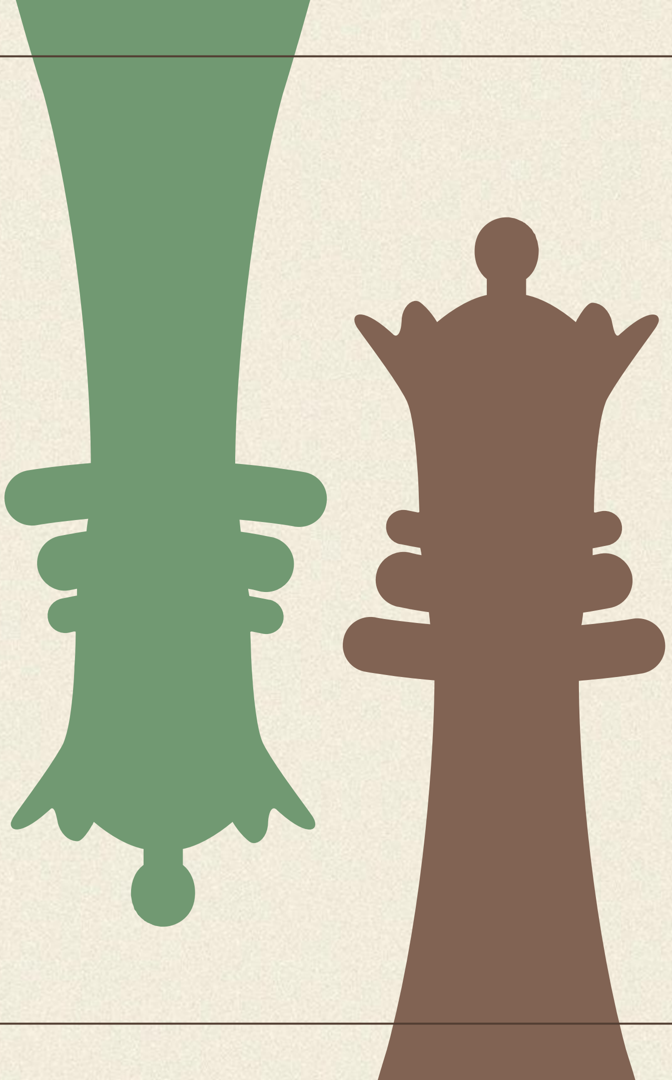


03

GREEDY APPROACH

GREEDY APPROACH

A Greedy algorithm is an algorithmic paradigm that builds up a solution piece by piece, always choosing the next piece that offers the most obvious and immediate benefit. So the problems where choosing locally optimal also leads to a global solution are best fit for Greedy . Depending on this algorithm , I used it to prepare a solution for my problem like the one of the Knapsack problem. I divided the attack points by the gold for each amount, after that I took the high possible ratios. For each type I just included one piece.

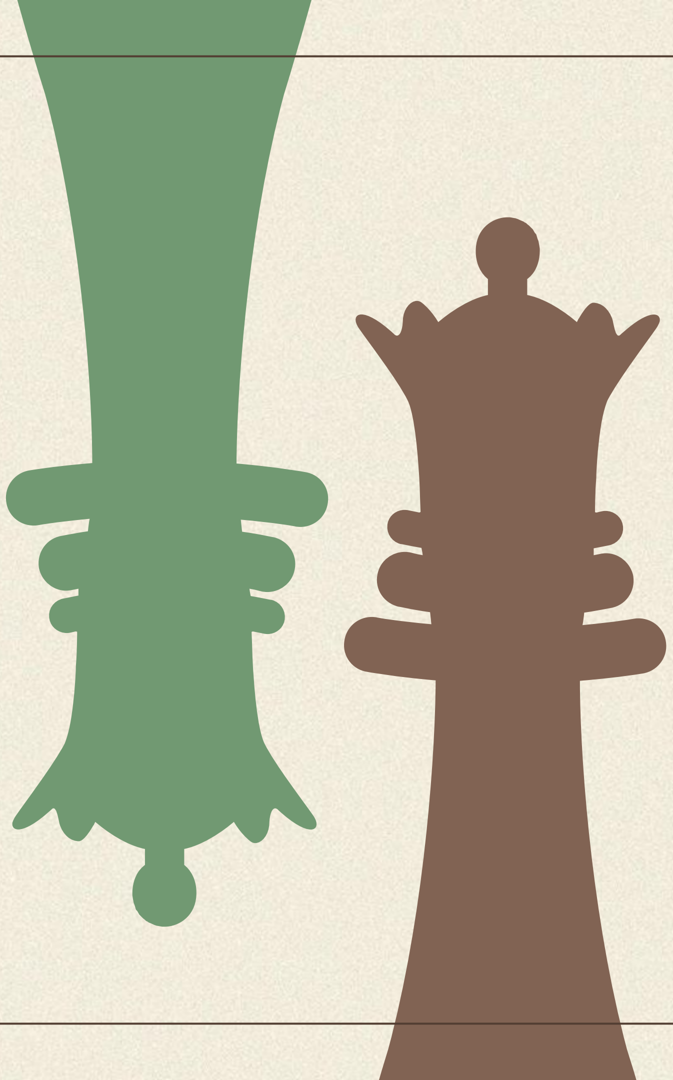


04

RANDOM APPROACH

RANDOM APPROACH

In this approach I just generate a random number in range of size of the list of selected pieces. As long as the Gold amount does not equal zero , still including the possible pieces , where I just include a one piece for each type.



05

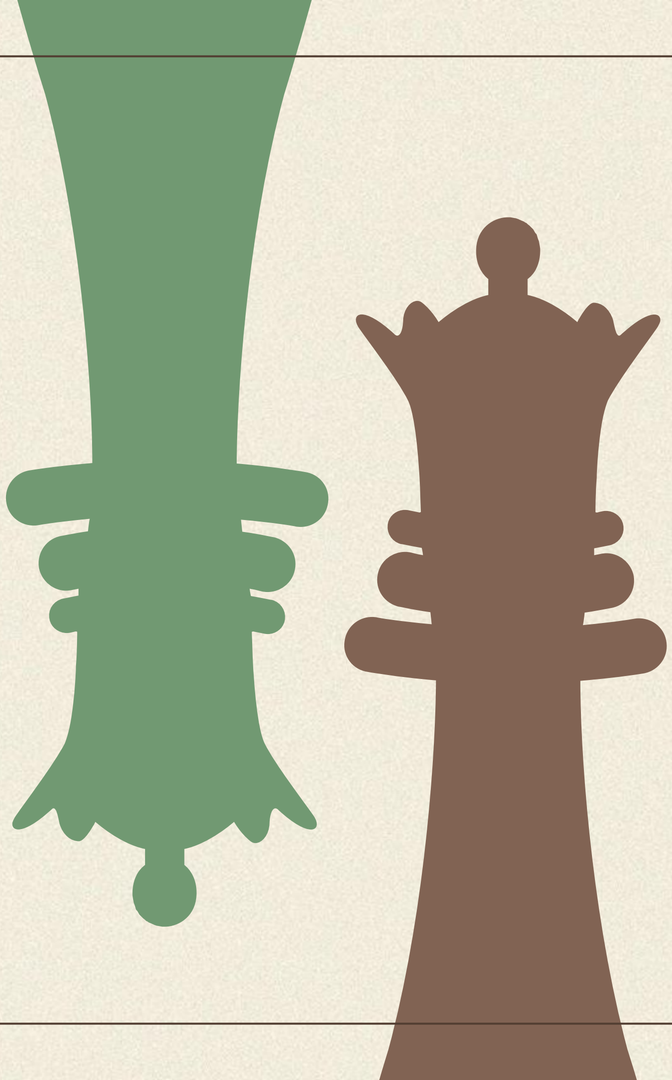
TIME TABLE

TIME TABLE

GREEDY	DYNAMIC	RANDOM
26177500 nsec	4987400 nsec	961600 nsec

** the above results are depending on the inputs:

- gold amount: 1000
- max level allowed: 5
- number of available piece per level: 5



06

CONCLUSION

CONCLUSION

This assignment was to have a good idea how to follow the appropriate technique to solve a problem. Dynamic and Greedy algorithms are a good solution for problems that required a lot of time and memory , where we can save the current results for each subproblems and use it again ,so that means less memory usage.

RESOURCES

- https://www.geeksforgeeks.org/
- https://www.youtube.com/watch?v=oTTzNMHM05I&ab_channel=AbdulBari
- https://en.wikipedia.org/wiki/Dynamic_programming
- https://www.javatpoint.com/dynamic-programming