University of Liverpool

Computer Science with a Year in Industry BSc (Hons)

G403

COMP390 Honours Year Computer Science Project

Design Specification

Author:
N Aishah B M Senin
(200912462)

Project Advisor:
Dr Prudence Wong

November 14, 2015

Contents

Chapter 1

Overview

1.1 Project description

This project primarily focuses on the animation of different types of commonly used algorithms, for the benefit of users to further understand how algorithms work in general. The scope of this project is within the animation of the main algorithmic paradigms, divide and conquer, greedy method, and dynamic programming.

Learning about what algorithms are and how they work is essential for students who are studying computer science. Since this project is meant to be educational, the target audience of the software will be students studying computer science, or at least have an interest on how computer programs are made efficient.

This project is to develop a software that displays animations that shows how an algorithmic solution works in general. From the program, the users are able to pick the algorithmic solution they wish to learn, enter a certain amount of input, or generate random values, and then learn how the algorithm works by watching the animations presented to them.

1.2 Aims and objectives of this project

The primary objective to this project is simply to make difficult algorithms easily understood. Also, using of visual aid as part of the educational process, for instance animations, to enhance the users' learning experience, which will make the students to learn new algorithmic problems with convenience and ease.

It is generally known the algorithms is one of challenging topics within the computer science field that is difficult for students to grasp on. So, the aim for this project is to allow students to achieve greater understanding in algorithmic paradigms, by providing an animated explanation in a step by step basis. To achieve this, the animation is to allow further speculation on how it works step by step, by breaking it down into smaller parts. This strategy of scrutinizing the algorithm allows the users to speculate the complicated algorithms in its granulated state, on how it works in each step, and then making a connection between the sequence of steps that makes the algorithm work as a whole.

Another aim for this project, is to provide the basic idea of how an educational program is suppose to look and work like in order to successfully assist the students. As a computer science student myself, I understand what are the specific difficulties when it comes to learning algorithms, and using them to address every difficulty I had when learning algorithms for the first time. Hopefully, once this project is completed, it will show the other developers who are interested in taking on this project on the specific areas to pay attention to when developing an educational program like this one.

I have also intend to serve this program as a base, where other developers could use to iterate from, by populating the list of available algorithms, by adding other algorithms into the list. If the project is deemed successful, universities could use this program to assist other students who are studying algorithms, or have difficulty understanding the concept of them.

Another aim for this project that would be nice to achieve, other than to benefit the students learning process, is to increase the students' interest on this topic. Algorithms is one of my favourite topics I had came across as a computer science student during my course in university. By designing and developing this program, I hope to achieve the same sentiments in regards to my interest in algorithms to other students who are studying this topic as well.

1.3 Summary of research and analysis

Do this

Chapter 2

Design

2.1 Functional and non-functional requirements

Not sure if saved settings belong in non-function

Table 2.1: Functional requirements of the software

No.	Requirements	Description
Menu		
1	Shows the list of playable algorithms	In the menu, the program is to show all the algorithms available in the program in the main list. In this list, user can select whichever algorithm they wish to see.
2	Classify the available algorithms between the 3 main algorithmic paradigms	On the main list, the algorithms are to be classified between the 3 main paradigms, such as the greedy method, divide and conquer, and dynamic programming. This is to allow the users to understand immediately the correlation between similar algorithms when classified within its paradigms. This is also to increase the ease of usability, as users will only be required to look within the algorithms paradigm to search for a specific problem.
Animation		
3	Plays the animation	When the animation is in its initial or paused state, users can play the animation. This initiates the animation, which plays until the end, unless the user either pauses or stops the animation.
4	Pauses the animation	The user can pause the animation, which stops the animation temporarily at its current state.
5	Stops the animation	When the animation is playing, user can stop the animation. This ends the animation completely at any point of time during the playtime of the animation.
6	Backtracks the animation	During the animation's playtime, the program keeps track on the number of iteration(s) the animation is currently at. When a user chooses to backtrack the animation, the animation will *rewind* itself from its current iteration *i*, to *i - 1*.
7	Shows a short description during the animation on each *iteration* of the algorithm	During the animation's playtime, the program is to show a short description about what the animation is doing.

Table 2.2: Functional requirements of the software

No.	Requirements	Description
Help option		
8	Adjust the speed of the animation	Users can adjust the speed of the animation ranging from 1 (very slow), to 10 (very fast). By default, the speed of the animation will be set to 5.
9	Adjust the font size	Users can adjust the font size to fit their own requirements. Users can pick sizes from small (font size 8), default (font size 12), and large (font size 16). By default, the general size of the fonts in the program will be sized 12.
Additional features		
10	Suggests to play similar algorithms	When users view a certain algorithm, the program also suggests an algorithm alike with the currently viewed one. This is to enhance better learning experience for users to seek out on similar problems
11	Appendix that shows further writeup of the algorithms available in the program	This shows the full writeup of the description shown during the animation, and additional information in regards with the algorithm.

Table 2.3: Non-functional requirements of the software

No.	**Requirements**	**Description**
Graphical interface		
1	The images for the animation is to be scalable depending on the size of the user's input	The physical size of the animation highly depends on the input size given by either the user or the random generator. Due to this, the program needs to carefully scale the animation when it is either too small or too big for the screen. It needs to ensure that the user can easily see the images and fonts of the animation, whether the input size is small or large.
2	Tables included in the animation demonstration are to be scrollable when it gets larger than a specified size given	Some algorithms require a table, especially the dynamic programming types. The table varies in size depending on the size of input for the algorithm. If the table width and length gets bigger than a specific size given, instead of exceeding the size, the program is to add a scrollable feature for the table.
3	The program is to be clear and easy enough for users to comprehend its design	The colour scheme of the program is to have a calming, non-blaring proposition. The images and fonts along with it needs to be shown clearly, and easily relatable for the general public.
Settings		
4	Saves the settings provided by user	The program is to save the changes made by user under settings. This means that when the user opens the program again, the changed settings will still be in placed.

Todo list