Learning Guide Unit 8

Site: University of the People
Course: CS 3304-01 Analysis of Algorithms - AY2024-T5
Book: Learning Guide Unit 8

Printed by: Mejbaul Mubin
Date: Saturday, 29 June 2024, 6:45 AM

Description

Learning Guide Unit 8

Table of contents

Overview

Unit 8: Limits to Computation (Part 2)

Topics:

- The Theory of NP-Completeness
- Impossible problems
- The Halting Problem

Learning Objectives:

By the end of this Unit, students will be able to:

1. Recognize and be able to Describe reductions
2. Define the halting problem and be able to articulate why this is important in Algorithm analysis
3. Define what a 'hard' problem is from the perspective of algorithm analysis
4. Identify problems that are NP complete

Tasks:

- Read the Learning Guide and Reading Assignments
- Participate in the Discussion Assignment (post, comment, and rate in the Discussion Forum)
- Make entries to the Learning Journal
- Take the Self-Quiz
- Read the Unit 9 Learning Guide carefully for instructions on the Final Exam
- Take the Review Quiz
- Complete and submit the anonymous Course Evaluation

Introduction

In unit eight we will continue our study of the theory of NP completeness.

NP Completeness
NP completeness or the theory of NP completeness is an approach to 'hard' problems.

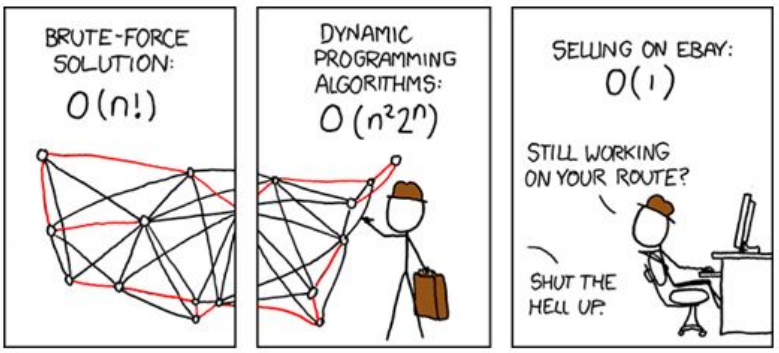
The most notable characteristic of NP-complete problems is that no fast solution to them is known; that is, the time required to solve the problem using any currently known algorithm increases very quickly as the size of the problem grows.

As a result, the time required to solve even moderately large versions of many of these problems quickly becomes so great that it isn't practical to solve them using current technology solutions.

Determining whether or not it is even possible to solve these problems quickly is one of the principal unsolved problems in Computer Science today.

While a method for computing the solutions to NP-complete problems using a reasonable amount of time remains undiscovered, computer scientists and programmers still frequently encounter NP-complete problems. It is important for programmers to be able to recognize an NP-complete problem so that they don't unknowingly waste time trying to solve a problem which so far has eluded generations of computer scientists.

The following url provides a humorous look at NP-complete problems. http://xkcd.com/399/



One approach that is taken to come up with a solution to NP-complete problems is to use techniques of approximation. If you recall the knapsack problem from unit 6, we had an example of a brute force approach that would look at every possible combination of items to put into a knapsack to get the 'optimal' contents which maximized the value of the items the thief would steal. In NP-complete problems this approach to finding an optimal solution is not practical so another approach might be to make an educated guess at the best solution and then test it to determine if in fact that it was the correct one.

Impossible Problems
In the final section of this course we will look at the so called impossible problems. One of the characteristics of an algorithm that we learned about in CS1303 was that an algorithm has a finite number of steps. It processes all of the data and when it has completed the processing ... stops. Unfortunately it is sometimes difficult to determine if in fact a particular algorithm will have a finite number of steps. Given the fact that some problems have a virtually infinite number of potential inputs it is virtually impossible (and in some cases impossible) to determine if the algorithm will in fact complete in a finite number of steps or if it will be an infinite loop. This situation is known as the halting problem and any problem that produces a solution that cannot be completed in a finite number of steps is an impossible problem.

Reading Assignment

Topic 1: The Theory of NP-Complete
Chapter 17: Limits to Computation in A Practical Introduction to Data Structures and Algorithm Analysis by Clifford A. Shaffer.Read "lecture 21: NP-Hard Problems available at http://drona.csa.iisc.ernet.in/~gsat/Course/DAA/lecture_notes/jeff_nphard.pdf

Chapter 8 NP Complete Problems in Algorithms by S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani available at http://www.cs.berkeley.edu/~vazirani/algorithms/chap8.pdf

Chapter 9 Coping with NP Completeness in Algorithms by S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani available at http://www.cs.berkeley.edu/~vazirani/algorithms/chap9.pdf

Topic 2: Impossible Problems including the Halting Problem
Chapter 17: Limits to Computation Section 17.1 in A Practical Introduction to Data Structures and Algorithm Analysis by Clifford A. Shaffer.

Discussion Assignment

Describe in your own words what the halting problems is and why it is relevant to the design and analysis of algorithms?

Include one or two examples to explain your thought process to show what is occurring and how the methodology works. Demonstrate your understanding of the intricacies of the halting problem and its influence on algorithms. Use APA citations and references for any sources used.

Learning Journal

The Learning Journal is a tool for self-reflection on the learning process. In addition to completing directed tasks, you should use the Learning Journal to document your activities, record problems you may have encountered and to draft answers for Discussion Forums and Assignments. The Learning Journal should be updated regularly (on a weekly basis), as the learning journals will be assessed by your instructor as part of your Final Grade.

Your learning journal entry must be a reflective statement that considers the following questions:

- Describe what you did. This does not mean that you copy and paste from what you have posted or the assignments you have prepared. You need to describe what you did and how you did it.
- Describe your reactions to what you did
- Describe any feedback you received or any specific interactions you had. Discuss how they were helpful
- Describe your feelings and attitudes
- Describe what you learned

Another set of questions to consider in your learning journal statement include:

- What surprised me or caused me to wonder?
- What happened that felt particularly challenging? Why was it challenging to me?
- What skills and knowledge do I recognize that I am gaining?
- What am I realizing about myself as a learner?
- In what ways am I able to apply the ideas and concepts gained to my own experience?

Your Learning Journal should be a minimum of 500 words

Self-Quiz

The Self-Quiz gives you an opportunity to self-assess your knowledge of what you have learned so far.

The results of the Self-Quiz do not count towards your final grade, but the quiz is an important part of the University's learning process and it is expected that you will take it to ensure understanding of the materials presented. Reviewing and analyzing your results will help you perform better on future Graded Quizzes and the Final Exam.

Please access the Self-Quiz on the main course homepage; it will be listed inside the Unit.

Review Quiz

The Review Quiz will test your knowledge of all materials learned in this course. The results of this quiz will not count towards your final grade, but will help guide you in more thoroughly reviewing necessary topics and adequately preparing for the Final Exam.

Please access the Review Quiz under Unit 9 on the main course homepage; it will be listed inside the Unit.

Checklist

- [ ] Read the Learning Guide and Reading Assignments
- [ ] Participate in the Discussion Assignment (post, comment, and rate in the Discussion Forum)
- [ ] Make entries to the Learning Journal
- [ ] Take the Self-Quiz
- [ ] Read the Unit 9 Learning Guide carefully for instructions on the Final Exam
- [ ] Take the Review Quiz
- [ ] Complete and submit the anonymous Course Evaluation