Project 4: Implementing Algorithms

Spring 2024 CPSC 335 - Algorithm Engineering
Instructor: Prof. Sampson Akwafuo

Abstract

In this project, you will develop pseudocodes for the algorithms; analyze your pseudocodes mathematically; implement the codes for the algorithm of your choice; test your implementation; and describe your results.

Problem 1: Rentals

A list of time intervals during which students at California State University, Fullerton needs a laptop is given to you. These time intervals are represented by pairs of integers $[start, end]$, where $0 <= start < end$. However, start and end do not represent real times; therefore, they may be greater than 24 . No two students can use a laptop at the same time. When a student is done using a laptop, another student can use that same laptop. For example, if one student rents a laptop during the time interval $[0, 2]$, another student can rent the same laptop during any time interval starting from 2 .
Design an algorithm that returns the minimum number of laptops that the school needs to rent such that all students will always have access to a laptop when they need one.

Sample Input

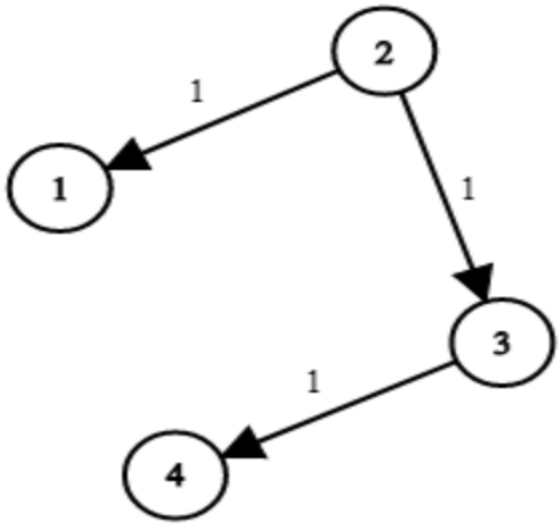
```
start_end = [
        [0, 2],
        [1, 4],
        [4, 6],
        [0, 4],
        [7, 8],
        [9, 11],
        [3, 10],
        ]
```

Sample Output = 3

Problem 2: Network Delay Time

Assume that you are given a network of **n** nodes, labeled from 1 $to\ n$. You are also given times: a list of travel times as directed edges times[i] = ***(u_i, v_i, w_i),*** where u_i is the source node, v_i is the target node, and w_i is the time it takes for a signal to travel from source to target.

A signal will be sent from a given node ***k***. Design an algorithm to return *the minimum time it takes for all the **n** nodes to receive this signal*. If the signal cannot be received by the ***n*** nodes, return -1.



Sample 1(Refer to the graph above graph):
Input: times = [[2,1,1],[2,3,1],[3,4,1]], n = 4, k = 2
Output: 2

Sample 2:
Input: times = [[1,2,1]], n = 2, k = 1
Output: 1

Sample 3:
Input: times = [[1,2,1]], n = 2, k = 2
Output: -1

To Do

1. Produce a written project report ***in PDF format***. Your report should include:
    a. Your name(s), CSUF-supplied email address(es), and an indication that the submission is for project 4.
2. Develop the pseudocode for Problems 1 and 2, and implement the **two** algorithms in Python or C++
3. Your codes should be saved and submitted in the corresponding executable file.
4. Mathematically analyze each algorithm and state the big *O* efficiency class.

Grading Rubric

The suggested grading rubric is given below:

Algorithm (55 points each)

a. Clear and complete Pseudocode = 10 points
b. Mathematical analysis and correct Big O efficiency class = 5 points
c. Inclusion of a Readme file = 5 points
d. Well commented codes = 5 points
e. Successful compilation of codes= 20 points
f. Produces accurate results = 10 points

Algorithm 2 (45 points)

a. Clear and complete Pseudocode = 10 points
b. Mathematical analysis and correct Big O efficiency class = 5 points
c. Inclusion of a Readme file = 5 points
d. Well commented codes = 5 points
e. Successful compilation of codes= 15 points
f. Produces accurate results = 5 points

Submitting Your Solutions

Submit your files to the Project 3 Assignment on Canvas. It allows for multiple submissions. Your codes should be submitted in their executable extensions (.py or .cpp), and report in PDF. All files should be submitted **separately**. Do not zip or use .rar.

Ensure your submissions are your own works. Be advised that your submissions may be checked for plagiarism using automated tools. Do not plagiarize. As stated in the syllabus, a submission that involves academic dishonesty will receive a 0% score on that assignment. A repeat offense will result in an "F" in the class and the incident will be reported to the Office of Student Conduct.

Deadline

The project deadline is **Wednesday, May 8, by 11:59 pm** on Canvas.
Penalty for late submission (within 48 hours) is as stated in the syllabus. Projects submitted more than 48 hours after the deadline will not be accepted.