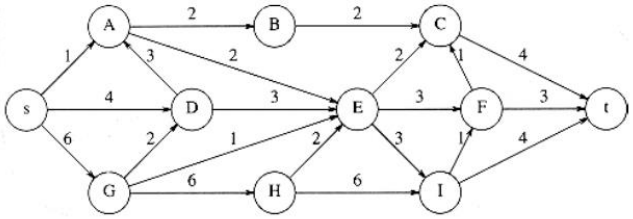
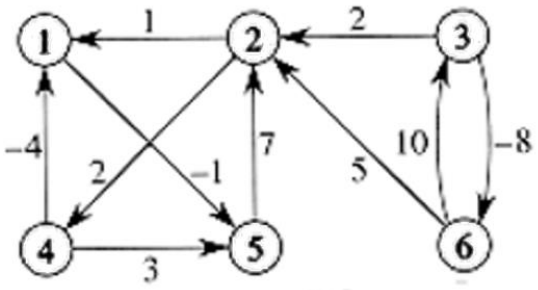
Concordia University
Dept. of Computer Science & Software Engineering
Asg 4 COMP 5511/2 Fall 2023 Due date: See CrsMgr
B. C. DESAI
https://confsys.encs.concordia.ca/CrsMgr

Theoretical Questions

- 1. [10 Points] Find a topological ordering for the graph in the following figure:.



- 2. [10 Points]Find all pairs of shortest distance for the following graph.



- 3. [20 Points] In this assignment, you are to create a Huffman code for the characters(lower case only) and numbers(0-9) in a sample text file. You could use the text file "The_State_of_Data_Final.txt[1]". To count the frequency of characters you can a write program (submit it) or use a script such as the grep Unix command given below and also in the Lab section. Draw the Huffman code tree for this frequency distribution and compare it with a binary tree for the same frequencies. Count the number of bits required to encode these characters in the sample text file for each of the two encoding.

```
grep -o The_State_of_Data_Final.txt -e a -e b -e c -e d -e e -e f -e g -e h -e i -e j \\
-e k -e l -e m -e n -e o -e p -e q -e r -e s -e t -e u -e v -e w -e x -e y -e z -e 0 \\
-e 1 -e 2 -e 3 -e 4 -e 5 -e 6 -e 7 -e 8 -e 9 -e - | sort | uniq -c
```

- 4. [10 Points] Let G be a graph with vertices A .. H. The adjacency of the vertices are given in the table below below. Assume that in a traversal, the adjacent vertices of a given vertex are in the order given in the following table. For example if you arrive at a node N (for which assume that the adjacency list is M, Q, R) the BFS would consider traversing to the nodes in the order M, Q, R! Draw G and give the sequence of vertices visited during a BFS and DFS traversal starting at A.

[1]To lower case characters in the text file, you can use in the emacs the command C-x C-l. Learn emacs if you do not know it!

A	**B, C, D, H**
B	**A, C, D**
C	**A, B, D, H**
D	**A, B, C, F**
E	**F, G, H**
F	**D, E, G**
G	**E, F, H**
H	**A, C, E, G**

Programming

Since the following two programming assignment requires some prepossessing of text files, write a module to do it[2]. This must be submitted as part of the assignment[3].

- 1. [50 Points] Write a program to read in a text file: the program removes all spaces and punctuations and convert the string to lower case. It than finds all LCS of the resulting strings from two files and prints them out. You may use the two text files[4] given in the Lab section of the course's home page in CrsMgr.
- 2. [50 Points] In this programming assignment, you will create a trie with all the non-noise word of the text of the document "The_State_of_Data_Final.txt". The intent is to locate paragraphs of the text of this document which has one or more search terms in it.

  The task involves, scanning a document in text format and eliminate all the "noise words"[5]. The non-noise words are trie keywords. The search for a set of keywords would give the paragraph(s) which contain these keywords.

  The program would start by reading a test document: you may, manually clean it up, but no textual paragraphs may be deleted. Parse it for words and eliminate stop words. For each keyword, your trie 'leaf' node would store the paragraphs numbers containing it. You need to store the paragraphs of the text only once.

  Once the entire text is parsed, create a trie index. Use this trie to perform a number of search tests. The test should include search involving one or more keywords in both conjunctive and disjunctive form[6].

  The response would be output of paragraphs satisfying the query.

  Your program stores all data in memory so all data structure would be stored in memory. You would have to devise and describe the data structures used in the system in the accompanying written report. Also give the input and the output. Note: your system could be tested on an alternate document.

[2]The software you develop must be general enough to use any two text files. You would be required to demonstrate the program with alternate text files to the Lab TAs.

[3]If your program gets exhausted, do at least a few pages of the document(s) instead of the whole!

[4]Two sample are: Privacy-in-the-Age-of-Information.txt and The_State_of_Data_Final.txt. Note: Since the text in these files is distilled from the corresponding files in different formats, some spaces between words may be missing and some characters may be mangled.

[5]Keywords are all non-noise words. One list (Some-noise-words.txt) is in the Lab section of the course home page in CrsMgr

[6]The result of the searches for at least the followings five types of query. There should be success and failure for each type. Query involving: keyword1; keyword1 and keyword2; keyworda or keywordb; keyword1 and (keyword2 or keyword3); keyword1 or (keyword2 and keyword3)

All groups are required to demonstrate the programming parts of this assignment to the Lab tutors in the week of Dec 4, 2023.

All submissions are to be uploaded to CrsMgr. CrsMgr will accept late submissions with a penalty per day; when the penalty reaches 100%, CrsMgr will no longer accept the submissions. Please do not email submissions to the instructor or the lab instructors.

Submission format: This is a group assignment: only one submission for each group is required: choose the best solution for each part and submit it as a single compressed tar-ball.

Peer evaluation: If the group is not of size one, make sure to evaluate the contribution of your peers. There is a peer evaluation for each group assignment and it must be done before the respective deadline.