School of Computer Science
University of Guelph

CIS*3490 The Analysis and Design of Algorithms

Winter 2022
Instructor: Fangju Wang

Assignment 5 (100%)

Question 1 (10%)

1.1 Construct an AVL tree for the list 79, 38, 42, 23, 18, 35. Insert the keys into the tree in the order of the list. For each key, show the tree with balance factors right after the key is inserted, and show the resultant tree with balance factors if a rotation is conducted.

1.2 Construct a 2-3 tree for the list 79, 38, 42, 23, 18, 35. Insert the keys into the tree in the order of the list. For each key, show the tree right after the key is inserted, and the resultant tree if a split operation is conducted.

Question 2 (15%)

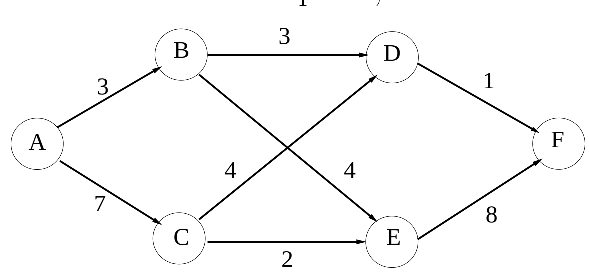
2.1 Construct a heap for the list 35, 42, 18, 79, 23, 38 in the array representation. Start with the original order of the numbers. Show intermediate steps in the heap construction. Please don't draw trees in your submission for this question.

2.2 Apply the method of heapsort to the heap constructed in 2.1, to sort the numbers in ascending order. Do the sorting in the array representation. Show intermediate steps. Please don't draw trees in your submission for this question.

2.3 Use the heap constructed in 2.1 as a priority queue, with the largest number being the object of highest priority. Remove the largest number from the queue, and then insert number 40 into the priority queue. Do the operations in the array representation. Show intermediate steps. Please don't draw trees in your submission for this question.

Question 3 (20%)

3.1 Apply the augmenting path algorithm to find the maximum flow in the following network. Break a tie using the alphabetical order. For each augmenting path you find, draw the network, highlight edges of the path, list vertices of the path, calculate the flow value of the path, and then update material amounts of the edges of the path, using the format of Figure 10.5 (page 364) in the text. When you have found all the paths, calculate the maximum flow.



3.2 Apply the shortest augmenting path algorithm to find the maximum flow in the network used in 3.1. Break a tie using the alphabetical order. For each augmenting path you find, draw the network, label the vertices, hightlight the edges of the path, list vertices of the path, calculate the flow value of the path, and then update material amounts of the edges of the path, using the format of Figure 10.7 (page 368) in the text. When you have found all the paths, calculate the maximum flow.

Question 4 (20%)

4.1 For each question in the following, find which item is NOT true. (Choose the most appropriate answer.)

1. The knapsack problem
   (a) is an NP-complete problem.
   (b) is an NP problem.
   (c) is a P problem because it has a dynamic programming algorithm of $\Theta(nw)$ where n is the number of items and w is the capacity of knapsack.
   (d) can be polynomially reducible to an NP-complete problem.

2. $P \quad = \quad NP$
   (a) holds if a P algorithm has been found for an NP-complete problem.
   (b) implies that we have a P algorithm for every NP problem.
   (c) has been disproved by a scientist at HP.
   (d) can be proved by finding a P algorithm for the CNF-satisfiability problem.

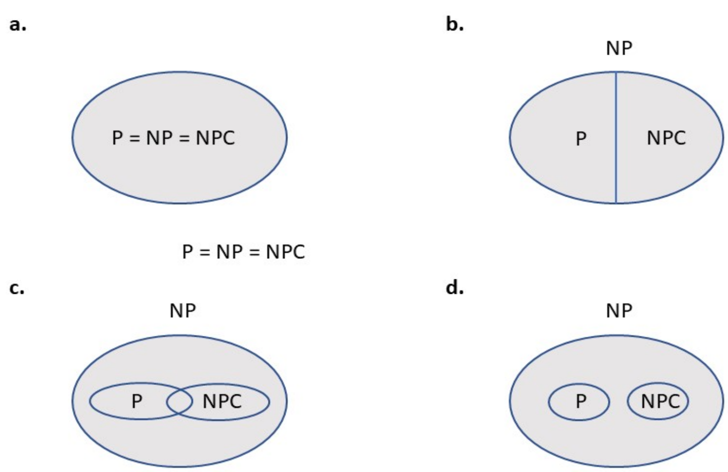
3. An *NP* complete problem can be solved by
   (a) an exponential or factorial algorithm.
   (b) a deterministic polynomial algorithm.
   (c) a nondeterministic polynomial algorithm.
   (d) an algorithm that consists of a guess stage and a verification stage.

4. An *NP* complete problem
   (a) is an *NP* problem.
   (b) is in *NP* and at least as difficult as any other problem in *NP*.
   (c) is an *NP* problem that has been completed.
   (d) is an *NP* problem and any other problem in *NP* can be reduced to it in polynomial time.

5. The class *NP* includes the decision versions of:
   (a) assignment problem.

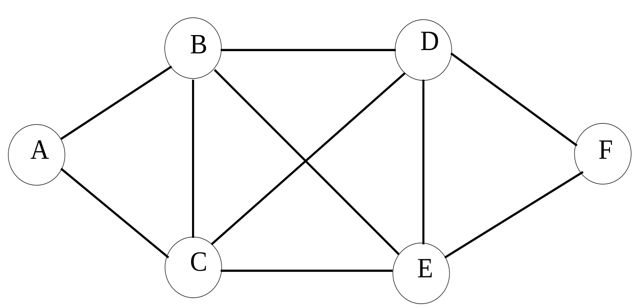
(b) traveling salesman problem
(c) Fibonacci number calculation
(d) bin packing.

4.2 Which of the following diagrams do not conflict the current state of our knowledge about the complexity classes P, NP, and NPC (NP-complete problem)? That is, which of the diagrams are possible, and which are impossible? Explain your answer for each diagram using one or two sentences.

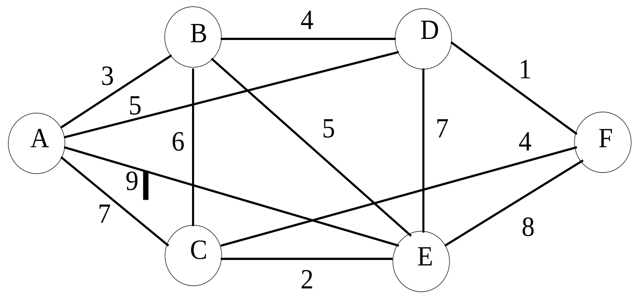


Question 5 (20%)

5.1 Apply backtracking to the problem of finding a Hamiltonian circuit in the following graph, starting and ending at vertex A. Break a tie using the alphabetical order. Draw your state-space tree in the format of Figure 12.3 (page 427).



5.2 Apply the branch-and-bound algorithm to solve the traveling salesman problem for the following graph. The salesman starts and ends at vertex A. Break a tie using the alphabetical order. Draw your state-space tree in the format of Figure 12.9 (page 439).



Question 6 (15%)

6.1 The following is an instance of the knapsack problem, where knapsack capacity is 11. Use the dynamic programming method to solve this knapsack problem. Show your table using the format of Figure 8.5 (page 294).

Item	*weight*	*value*
1	7	\$42
2	3	\$12
3	4	\$40
4	5	\$25

6.2 Use the Sahni's approximation scheme to solve the knapsack problem in 6.1 for $k = 2$. Show your process using the format of Figure 12.16 (b) (page 457). What is the accuracy ratio of your solution?

Due time: 08:00, Monday, April 11, 2022. Please submit an e-copy to Moodle.