EAST WEST UNIVERSITY

Department of Computer Science and Engineering

Lab Manual: CSE245/246 Algorithms (Section: All)

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: Divide and Conquer Lab: 01

1. Binary search: Given a sorted array of n integers and a target value, determine if the target exists in the array in logarithmic time using the binary search algorithm. If target exists in the array, print the index of it.

Sample input	Sample output
Data: 2, 3, 5, 7, 9 Target: 7	Found at index: 3
Data: 6, 7, 12 Target: 15	Not found

2. Merge Sort: Given an integer array, sort it using the merge sort algorithm.

Sample input	Sample output
Data: 2, 3, 7,5	2 3 5 7
Data: 12, 6, 7	6 7 12

3. Quick Sort: Given an integer array, sort it using the merge sort algorithm.

Sample input	Sample output
Data: 2, 3, 7,5	2 3 5 7
Data: 12, 6, 7	6 7 12

4. Closet pair of points: We are given an array of n points in the plane, and the problem is to find out the closest pair of points in the array.

Sample input	Sample output
{2, 3}, {12, 30}, {40, 50}, {5, 1}, {12, 10}, {3, 4}	The smallest distance is 1.41421

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: Greedy approach Lab: 02

1. Fractional knapsack: Given the weights and profits of N items, in the form of {profit, weight} put these items in a knapsack of capacity W to get the maximum total profit in the knapsack. In Fractional Knapsack, we can break items for maximizing the total value of the knapsack.

Sample input	Sample output
{60, 10} {100, 20} {120, 30} W = 50	240

2. Activity selection problem: You are given n activities with their start and finish times. Select the maximum number of activities that can be performed by a single person, assuming that a person can only work on a single activity at a time.

Sample input	Sample output
start = {10, 12, 20} finish = {20, 25, 30}	2

3. Scheduling problem: Given a schedule containing arrival and departure time of trains in a station, find minimum number of platforms needed in the station so to avoid any delay in arrival of any train.

Sample input	Sample output
Number of schedules: 6 Arrival: 2.00 2.10 3.00 3.20 3.50 5.00 Departure: 2.30 3.40 3.20 4.30 4.00 5.20	2

4. Job sequencing Problem: Given an array of jobs where every job has a deadline and associated profit if the job is finished before the deadline. It is also given that every job takes a single unit of time, so the minimum possible deadline for any job is 1. Maximize the total profit if only one job can be scheduled at a time.

Sample input	Sample output
a 4 20 b 1 10	c, a

c	1	40	
d	1	30	

5. Job sequencing problem – Loss minimization: We are given N jobs numbered 1 to N. For each activity, let Ti denotes the number of days required to complete the job. For each day of delay before starting to work for job i, a loss of Li is incurred. You are required to find a sequence to complete the jobs so that overall loss is minimized. You can only work on one job at a time.

Sample input	Sample output
L = {3, 1, 2, 4} T = {4, 1000, 2, 5}	3, 4, 1, 2

6. Assign mice to hole: There are N Mice and N holes are placed in a straight line. Each hole can accommodate only 1 mouse. A mouse can stay at his position, move one step right from x to x + 1, or move one step left from x to x -1. Any of these moves consumes 1 minute. Assign mice to holes so that the time when the last mouse gets inside a hole is minimized.

Sample input	Sample output
positions of mice are: 4 -4 2 positions of holes are: 4 0 5	4

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: Number theory Lab: 03

1. GCD: Given two numbers a and b, the task is to find the GCD of the two numbers using Euclid's algorithm.

Sample input	Sample output
a = 20 b = 28	4

2. Prime factor: Given a number n, write an efficient function to print all prime factors of n. For example, if the input number is 12, then the output should be "2 2 3". And if the input number is 315, then the output should be "3 3 5 7".

Sample input	Sample output
315	3 3 5 7

3. Sieve method: Given a number n, print all primes smaller than or equal to n using sieve method. It is also given that n is a small number.

Sample input	Sample output
10	2 3 5 7

4. Highest occurring digit: Given a range L to R, the task is to find the highest occurring digit in prime numbers lie between L and R (both inclusive). If multiple digits have the same highest frequency print the largest of them. If no prime number occurs between L and R, output -1.

Sample input	Sample output
L = 1 R = 20	1

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: String and Pattern matching Lab: 04

1. String matching: Implement a program to search for a pattern in a text using the naive string-matching algorithm.

Sample input	Sample output
Text: "The quick brown fox jumps over the lazy dog." Pattern: "fox"	Pattern found at index 16.

2. Rabin-Karp: Implement a program to search for a pattern in a text using the Rabin-Karp algorithm.

Sample input	Sample output
Text: "The quick brown fox jumps over the lazy dog." Pattern: "fox"	Pattern found at index 16.

3. Knuth-Morris-Pratt: Implement a program to search for a pattern in a text using the Knuth-Morris-Pratt algorithm.

Sample input	Sample output
Text: "The quick brown fox jumps over the lazy dog." Pattern: "fox"	Pattern found at index 16.

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: Dynamic Programming (Part-01) Lab: 05

1. 0-1 Knapsack: You are given a set of items, each with a weight and a value, and a knapsack with a maximum weight capacity. Your task is to determine the maximum value that can be obtained by selecting a subset of the items to fit into the knapsack without exceeding its weight capacity.

Sample input	Sample output
4 2 3 3 4 4 5 5 6 8	9

2. Sum-of-Subset: Given a set of positive integers and a target sum, your task is to determine whether there exists a subset of the given set whose elements sum up to the target sum.

Sample input	Sample output
5 1 3 5 7 9 12	Yes

3. Coin change: You are given n types of coins and another number K. Your task is to determine whether it is possible to generate K using those coins if
   i. The number of each coin is infinite.
   ii. The number of each coin is finite.

4. Coin change: You are given n types of coins and another number K. Your task is to find the minimum number of coins required to make K if.
   i. The number of each coin is infinite.
   ii. The number of each coin is finite.

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: Dynamic Programming (Part-02) Lab: 06

1. LIS: Given an array of integers, your task is to find the length as well as the sequence of the longest increasing subsequence within the array.

Sample input	Sample output
8 5 2 8 6 3 6 9 7	4 2, 3, 6, 9

2. LCS: You are given two strings, and your task is to find the length of the longest common subsequence (LCS) between them. Also print the LCS.

Sample input	Sample output
string1: "ABCDGH" string2: "AEDFHR"	3 “ADH”

3. Longest Common Substring: You are given two strings, and your task is to find the length of the longest common substring between them. Also find the substring itself.

Sample input	Sample output
string1: "ABCDGH" string2: "ACDGHR"	4 “CDGH”

4. Longest palindromic subsequence: You are given a string, and your task is to find the length of the longest palindromic subsequence (LPS) within the string.

Sample input	Sample output
string: "BBABCBCAB"	7 “BABCBAB”

East West University
Department of Computer Science and Engineering
Course: CSE246 Algorithm Topic: Graph Theory (BFS) Lab: 07

1. Shortest path: You are given an unweighted, undirected graph as input. Your task is to find the shortest path from source node to all other nodes using Breadth-First Search (BFS) algorithm. First input n denotes the number of nodes followed by number of edges. Then each edge is given as a pair of integer values (u, v).

Sample input	Sample output
4 4 0 1 1 2 2 0 2 3 Source node: 0	Path 0 to 1: 0->1 Cost: 1 Path 0 to 2: 0->2 Cost: 1 Path 0 to 3: 0->2->3 Cost: 2

2. Bipartite graph: You are given an unweighted, undirected graph as input. Your task is to determine whether the given graph is bipartite or not using Breadth-First Search (BFS) algorithm. First input n denotes the number of nodes followed by number of edges. Then each edge is given as a pair of integer values (u, v).

Sample input	Sample output
4 4 0 1 1 2 2 0 2 3	Not Bipartite

3. Maze solves: You are given a n*m 2D matrix as input. Each cell of the matrix will contain either 0 or 1. 0 means you can't go that cell and 1 means you can go there. You can move only up, down, right, and left by one cell only. For example, if your current position in matrix is 2, 2 then you can move only to (2, 3), (1, 2), (2, 1) and (3, 2) position if those positions contain 1. You are given a starting and an ending position as input. Now write a program using that find a path from starting position to end position using DFS algorithm. Consider the following example for better understanding.

1	0	0	0	1
1	0	0	0	1
1	1	1	0	0
0	0	1	1	0

0	1	1	0	0

Let 0, 0 is the starting position and 2, 2 is the ending location. So one possible path between 0, 0 and 2, 2 is (0, 0) -> (1, 0)-> (2, 0) -> (2, 1) -> (2, 2).