Name : SONAM
Sec : B
University Roll. : 2014892.

DATE / /
PAGE NO.

Tutorial 7

Q1. Greedy algo paradigm : Greedy is an algorithm paradigm that builds up a solution piece by piece, always choosing the next piece that offers the most obvious and immediate benefits.
There are multiple applications of the greedy technique such as:
1. CPU scheduling
2. Minimum spanning Tree
3. Several Graph based algo's.

Q2.

	Activity Selection	Job sequencing	Fractional Knapsack	Huffman Encoding
Time Complexity	O(N log N)	$O(n^2)$	O(n log n)	O(n log n)
space complexity	O(n)	O(n)	O(n)	O(n)

Q3 Huffman coding

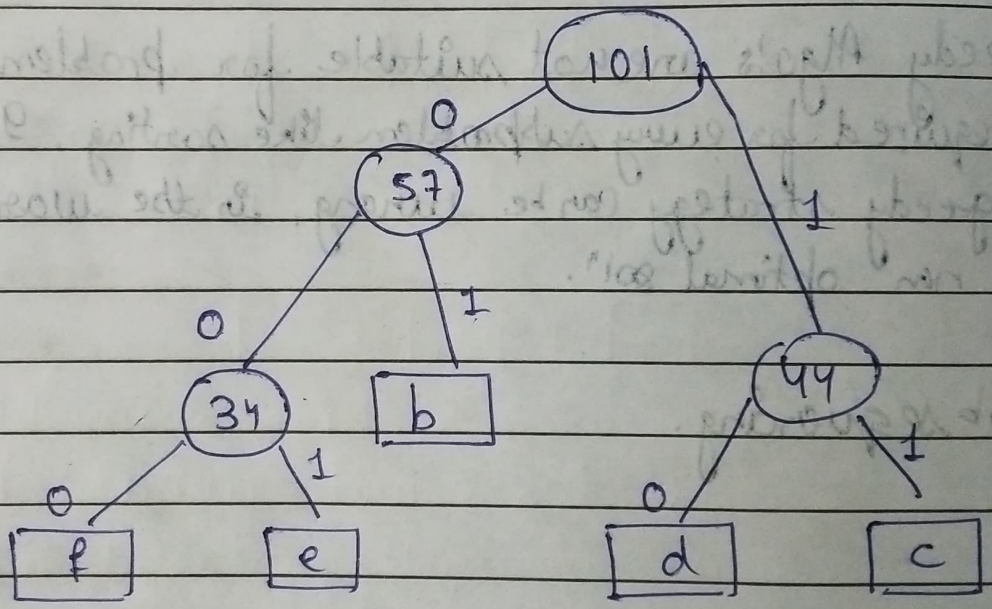
Char	Frequency
a	45
b	23
c	22
d	20
e	19
f	15

Sorted Frequency

15	19	20	22	23	45
f	e	d	c	b	a

Huffman tree.

~~f — 15~~
~~e — 19~~
~~d — 20~~
~~c — 22~~
~~b — 23~~
~~a — 45~~
~~fe — 34~~
~~dc — 44~~
feb - 57
febdc - 101



f - 000
e - 001
b - 01
d - 10
c - 11.

Q4 Data Structure used for Huffman Encoding: Binary Tree is used for building Huffman coding and it is also used for Huffman Encoding.

Application of Huffman encoding.

1. Huffman code is used to convert fixed length codes into variable length codes, which result in lossless compression
2. Compressed codes may be further compressed using JPEG and MPEG. to get the desired compression ratio

Q5. W=15

Value	10	5	15	7	6	18	3
Weight	2	3	5	7	1	4	1
V/W	5	1.6	3	1	6	4.5	3

- Choose highest V/W ratio for which weight $w \leq W$
- let current weight = c.

DATE / /
PAGE NO.

X = | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

c = 1

x = | 1 | 0 | 0 | 0 | 1 | 0 | 0 |

c = 3

x = | 1 | 0 | 0 | 0 | 1 | 1 | 0 |

c = 7

x = | 1 | 0 | 1 | 0 | 1 | 1 | 0 |

c = 12

x = | 1 | 0 | 1 | 0 | 1 | 1 | 1 |

c = 13.

x = | 1 | 2/3 | 1 | 0 | 1 | 1 | 1 |

next heightest weight is 1.6
which has 3 unit weight, but c + 3 ≯ 15.
Hence we will fraction it as per requirement.

required weight = W - C = 15 - 13 = 2.
Hence weight added is $\frac{2}{3}$.

xw = | 2 | 2 | 5 | 0 | 1 | 4 | 1 | $\Sigma x_i w_i = W$.

vx = | 10 | 3.3 | 15 | 0 | 6 | 18 | 3 |

Max profit / max value = $\Sigma v_i x_i$
= 55.

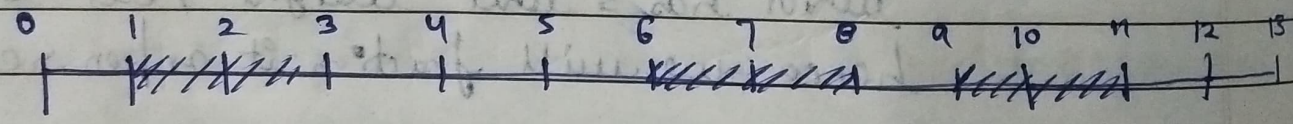
Q6 Knapsack Algo: To solve the problem we take the value/weight ratio and on the basis of this ratio a weight is, which has highest v/w ratio, added to the Knapsack. until we can't add the next weight as a whole and that point of time we take the required fraction of the weight and add it to the Knapsack. This is nothing but greedy approach of taking the highest ratio everytime.

Huffman Coding: Huffman coding is based on the frequency of the character. We assign the variable length code to input characters, length of the assigned codes are based on the frequencies of corresponding characters. Hence; it is a greedy approach as we are using a predefined structure everytime to solve the problem.

Q7

	a	b	c	d	e	f
Start time	1	2	0	6	9	10
End time	3	5	7	8	11	12

Timeline



included procces - a, d, e

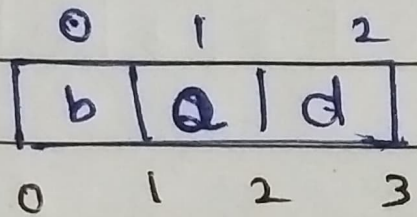
max no. of procces = 3.

DATE / /
PAGE NO.

Q8

	a	b	c	d	e
profits	20	15	10	5	1
Deadline	2	2	1	3	3
			X		

Time line.



profit 20 + 15 + 5 = 40.

Q9- Greedy Algo's are not suitable for problems where a soln is required for every subproblem like sorting. In such problems the greedy strategy can be wrong, in the worst can even lead to a non optimal soln.