NAME
"Zunair bin Ahmad"

Roll no
"BSCS-E1-19-14"

QNo1:- Knap Sack problem with weight of knap of even, odd.

You can keep two tables with n rows and C colums:

Dp even that saves the best Knapsack Solution with an even number of items.
Dp odd that saves the best Knapsack solution with odd number of items.

To fill DPeven you look at the previous best solution of DPodd plus a new item, or of DPeven if you don't take the item.

$$DP_{even}[i,j] = \max(DP_{odd}[i-1, j-c[i]] + v[i], DP_{even}[i-1, j])$$

The same idea goes for DPodd

$$DP_{odd}[i,j] = \max(DP_{even}[i-1, j-c[i]] + v[i], DP_{odd}[i-1, j])$$

Your solution will be in DPeven $[n, c]$

QNo2-

Huffman Codes

i):- Data can be encoded efficiently using huffman codes.

ii):- it is a widely used and beneficial technique for compressing data.

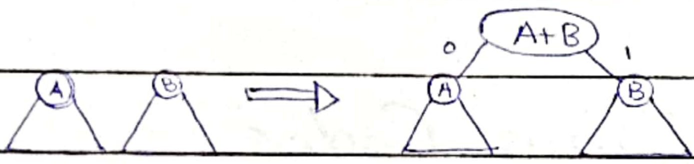
iii):- Huffman's greedy algorithm uses a table of frequencies of occurences of each character to build up an optimal way of representing each character as

Date:

a binary string.

Greedy Algorithm for Constructing a Huffman Code:-

Huffman invented a greedy algorithm that creates an optimal prefix called a Huffman code.



The algorithm builds the tree T analogous to the optimal code in a bottom-up manner.

Algorith of Huffman Code

Huffman(C)

1):- n = |C|

2):- Q ← C

3):- for i=1 to n-1

4):- do

5):- z = allocate-Node ()

6):- x = left [z] = Extract-Min (Q)

7):- y = right[z] = Extract-Min (Q)

Date:

8):- $f[z] = f[x] + f[y]$

9):- Insert (Q, z)

10):- return Extract-Min (Q)

Example:-

a: 50 , b: 25 , c: 15 , d: 40 , e: 75

Solution

Given that, $C = \{a, b, c, d, e\}$

$f(c) = \{50, 25, 15, 40, 75\}$

$n = 5$

$Q \leftarrow C$

for $i \leftarrow 1$ to 4

$i = 1$ $\quad z \leftarrow$ Allocate node

$x \leftarrow$ Extract - Min(Q)

$y \leftarrow$ Extract - Min(Q)

Left $[z] \leftarrow x$

Right $[z] \leftarrow y$

$f(z) \leftarrow f(x) + f(y) = 15 + 25$

$f(z) = 40$

Date:

Greedy algorithm of Huffman coding:-

~~Greedy-A~~

1):- $n \leftarrow$ length [S]

2):- $A \leftarrow \{1\}$

3):- $j \leftarrow 1$

4):- for $i \leftarrow 2$ to n

5):- do if $S_i \geq f_i$

6):- then $A \leftarrow A \cup \{i\}$

7):- $j \leftarrow i$

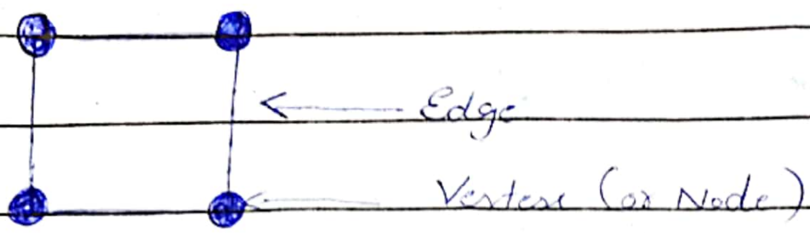
8):- return A

"Yes it is related to activity selection process.

The activity selection problem is a mathematical optimization problem."

Q No 3:-

Graph

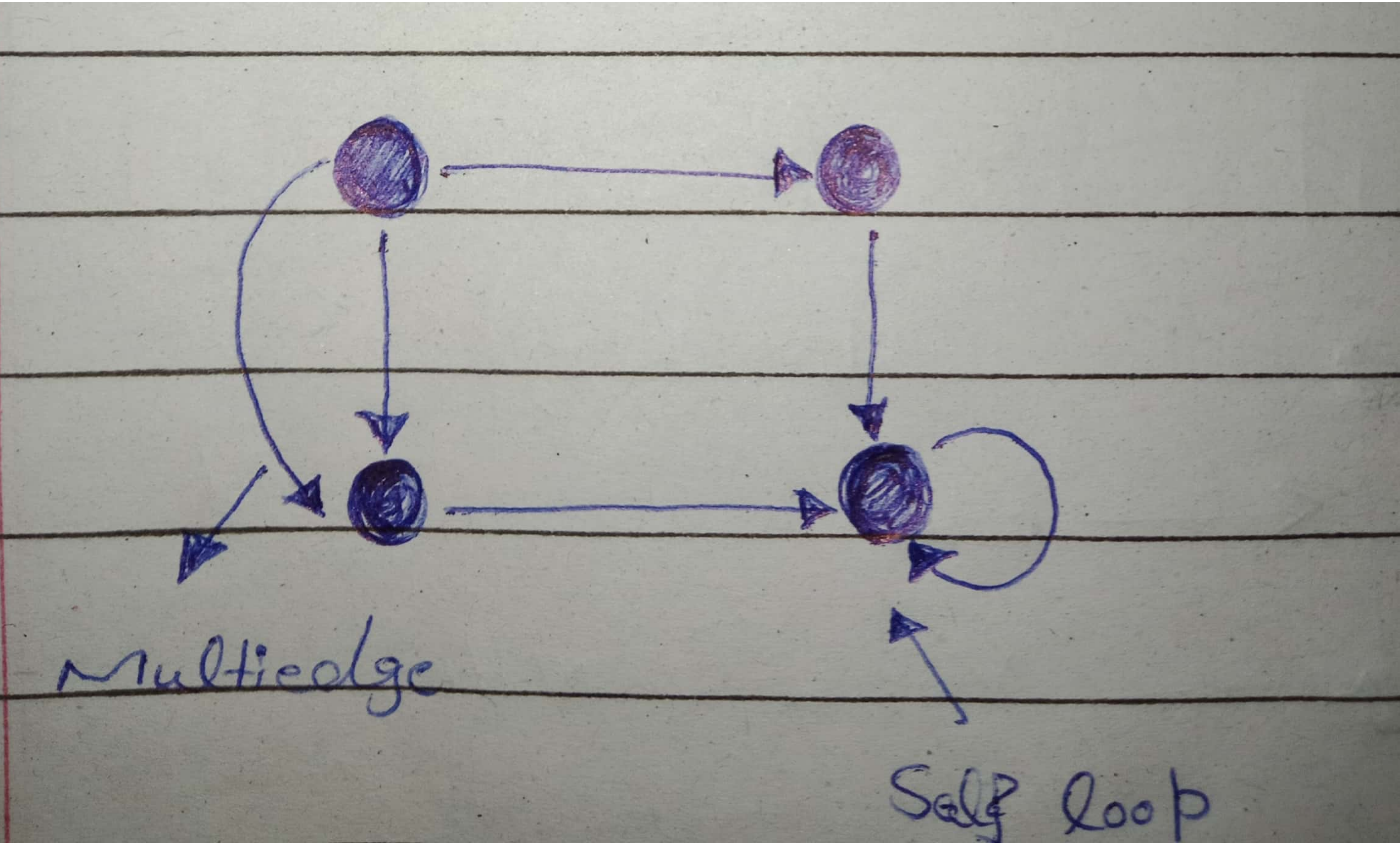


A graph

In simplest terms, a graph is a combination of verticles (or nodes) and edges.
We have 4 nodes and 4 edges and it is a graph.
Graph is a very important data structure to store data which are connected to each other.

Digraph with multiedge.

The graph which has self loops or an edge (i, j) occurs more than once (also called multiedge and graph is called multigraph) is a non-simple graph.



Multiedge
Self loop