Name → Hemant Dilip Tighade.

Roll No → 19CE1069

Sign → Hemant

Batch → B4.

Page → 1

Q.1)

i)

Greedy approach	Dynamic programming.
1. Greedy Method is used for obtaining optimum solution.	1) Dynamic programming is also for obtaining optimum solution
2. In greedy method a set of feasible solutions and the picks up the optimum solution.	2) There is no special set of feasible solution in this method.
3. In this method the optimum selection is without revising previously generated solutions	3) It considers all possible sequences in order to obtains the optimum solution.
4. There is no as such guaranteed of getting optimum solution	4) It is gauranteed that the dynamic programming will generate optimal solution using principle of optimality.
5. In this method only one decision sequence is generated	5) In this Method many decision sequences may be generated.

Name → Hemant Dilip Tighade

Roll No → 19CE1069 sign → Hemant

Batch → B4

page → 2

Q.1)

iii) - This ~~ex~~ is the simple and inefficient brute force approach. It compares first character of pattern with searchable text. If match is found pointers in both strings are advanced. If match is not found pointer of text is incremented and pointer of pattern is reset. This process is repeated till the end of text

- Naive approach does not require any pre-processing Given text T and pattern P, it directly starts comparing both strings character by character.

- After each comparison it shifts pattern string one position to the right

- following ~~explai~~ example illustrates the working of naive string matching algorithm, Here T = PLANINGANDANALYASIS and P = AND

Here ti and Pj are indices of text and pattern respectively.

Name → Hemant Dilip Tighade

Batch → B4 Sign → Hemant

Roll No → 19CE1069

page → 3.

Algorithm → NAIVE_STRING_MATCHING(T, P)

//T is the text string of length n

//P is the pattern of length m

for i ← 0 to n-m do

if P[1....m] == T[i+1...i+m] then

print "Match Found"

end

end.

Name → Hemant Dilip Tighade

Roll No → 19CE1069

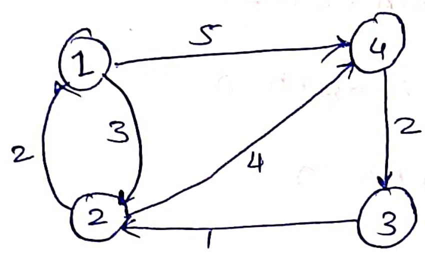
Sign → Hemant

Batch → B4

page → 4.

Q.2

i)



Solution -

$$D_k[i,j] = \min\{D_{k-1}[i,j],\ D_{k-1}[i,k] + D_{k-1}[k,j]\}$$

$$D_0 = \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & \infty & 4 \\ \infty & 1 & 0 & \infty \\ \infty & \infty & 2 & 0 \end{bmatrix} \qquad \pi_0 = \begin{bmatrix} 0 & 1 & \infty & 1 \\ 2 & 0 & \infty & 2 \\ \infty & 3 & 0 & \infty \\ \infty & \infty & 4 & 0 \end{bmatrix}$$

Iteration 1

$$D_1 = \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & \infty & 4 \\ \infty & 1 & 0 & \infty \\ \infty & \infty & 2 & 0 \end{bmatrix} \qquad \pi_1 = \begin{bmatrix} 0 & 1 & \infty & 1 \\ 2 & 0 & \infty & 2 \\ \infty & 3 & 0 & \infty \\ \infty & \infty & 4 & 0 \end{bmatrix}$$

Iteration 2

$$D_2 = \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & \infty & 4 \\ 3 & 1 & 0 & 5 \\ \infty & \infty & 2 & 0 \end{bmatrix} \qquad \pi_2 = \begin{bmatrix} 0 & 1 & \infty & 1 \\ 2 & 0 & \infty & 2 \\ 2 & 3 & 0 & 2 \\ \infty & \infty & 4 & 0 \end{bmatrix}$$

Name → Hemant Dilip Tighade

Roll No → 19CE1069

Batch → B4.

Sign → Hemant

page → 5.

Iteration 3

$$D_3 = \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & \infty & 4 \\ 3 & 1 & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix} \qquad \Pi_3 = \begin{bmatrix} 0 & 1 & \infty & 1 \\ 2 & 0 & \infty & 2 \\ 2 & 3 & 0 & 2 \\ 3 & 3 & 4 & 0 \end{bmatrix}$$

Iteration 4

$$D_4 = \begin{bmatrix} 0 & 3 & 7 & 5 \\ 2 & 0 & 6 & 4 \\ 3 & 1 & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix} \qquad \Pi_4 = \begin{bmatrix} 0 & 1 & 4 & 1 \\ 2 & 0 & 4 & 2 \\ 2 & 3 & 0 & 2 \\ 3 & 3 & 4 & 0 \end{bmatrix}$$

The final matrix with the sortest path distance between every pair is

$$D_4 = \begin{bmatrix} 0 & 3 & 7 & 5 \\ 2 & 0 & 6 & 4 \\ 3 & 1 & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix}$$

Name ⇒ Hemant Dilip Tighade

Roll No ⇒ 19CE1069

Batch → B4

page → 6

Sign → Hemant

III)

```
Algorithm nqueen(k,n)
//this procedure prints all possible
// placement of n queen on an n*n
// chess board so that they are
// non-attacking
{
 For i=1 to n do
 {
  if place(k,i) then
  {
   x[k]=i;
   if(k=n) then write(x[1:n]);
   else nqueen(k+1,n);
  }
 }
}
```

Name → Hemant Dilip Tighade

Roll No → 19CE1069

Batch → B4

page → 7

sign → Hemant

```
Algorithm place (k, i)
// return true if a queen can be placed in
// kth row ith column Else it return False
// X[] is a global array. abs (r) returns
// absolute value of r
{
    for j = 1 to k - 1 do
    if ((x[j] = i) // same column
    or (abs (x [j] - i) = abs (j - k)) // same diagonal
    then return False;
        return true;
}
```