UNIVERSITY OF ASIA PACIFIC

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING(CSE)

Course Title: Data Structures and Algorithms II Lab

Course Code: CSE 208

Report Name: Problem Statement (1-2) Assignment

Date of Submission: 1December2024

Submitted By,

Name: H.M. Tahsin Sheikh

ID: 22201243

Sec: E

Submitted To,

Fabliha Haque (Lecturer)

Problem Statement 01:

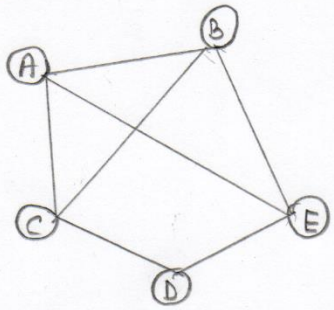
```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#define MAX 10
//Declaration of board size
int board[MAX];
int solutions = 0;
//Function to Check if Queen Position safe or not
bool isSafe(int row, int col)
{
   for (int i = 0; i < row; i++)   //Loop through Row level
   {
      //Checking Column and Diagonal to ensure Queen safe
      if (board[i] == col || abs(board[i] - col) == abs(i - row))
         return false; //Not safe
   }
   return true; //If True Queen is safe
}
//Print the Solution by this Function
void printSolution(int n)
{
   printf("Solution %d:\n", ++solutions);
   for (int i = 0; i < n; i++)
```

```
    {
        for (int j = 0; j < n; j++)
        {
            if (board[i] == j)
                printf("Q ");
            else
                printf(". ");
        }
        printf("\n");
    }
    printf("\n");
}
//using backtracking
void solveNQueens(int row, int n)  //If already Q placed then print current solve
{
    if (row == n)
    {
        printSolution(n);
        return;
    }
    for (int col = 0; col < n; col++)  //Try to place the Queen
    {
        if (isSafe(row, col))
        {
```

```
            board[row] = col;
            solveNQueens(row + 1, n);
        }
    }
}
int main()
{
    int n;
    //Size of N-Queen
    printf("Enter the value of N: ");
    scanf("%d", &n);
    if (n < 1 || n > MAX)
    {
        printf("Invalid input. Please enter N between 1 and %d.\n", MAX);
        return 1;
    }
//Output Part
    solveNQueens(0, n);
    printf("Total Solutions: %d\n", solutions);
    return 0;
}
```

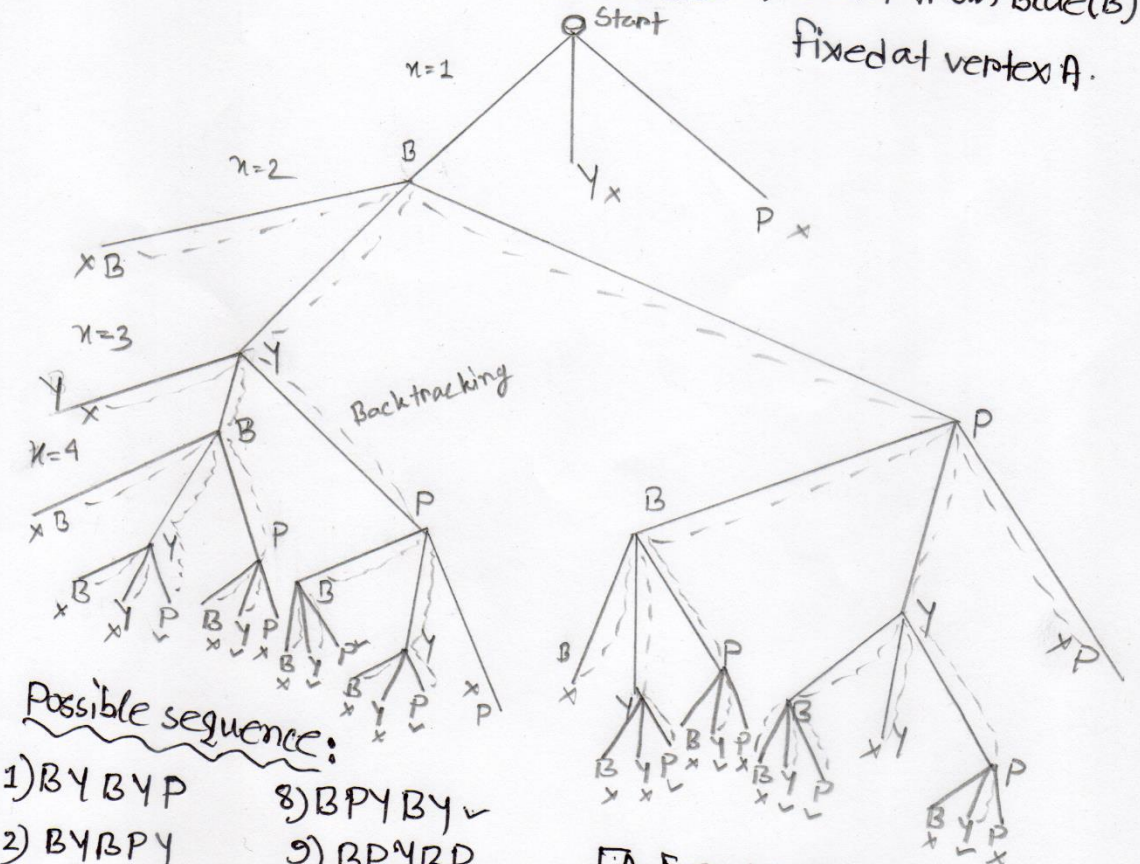
Problem Statement 02:

Problem Statement 2:



Hence,
vertex = 5
∴ n = 5

Since, ID is 22201243 ; 3 a odd number : start from Blue(B) fixed at vertex A.



Possible sequence:

1) BYBYP
2) BYBPY
3) BYPBY
4) BYPBP ✓
5) BYPYP ✓
6) BPBYP
7) BPBPY
8) BPYBY ✓
9) BPYBP
10) BPYPY ✓

[A, E and C, B is connected
So, only 4, 5, 8, 10 selected]

∴ Orginal sequence are,
BYPBP, BYPYP, BPYBY, BPYPY