S.No: 14	Exp. Name: ***Program to implement N-Queen's problem using backtracking***	Date:

Aim:

Program to implement N-Queen's problem using backtracking

Source Code:

nQueen.c

```
#include<stdio.h>
#include<conio.h>
int board[20],count;
int main()
{
   int n, i, j;
   void queen(int row, int n);
   printf("Enter number of Queens: ");
   scanf("%d",&n);   queen(1, n);
   return 0;
}
void print(int n)
{
   int i, j;
   for(i=1;i<=n;i++)
   {
      for(j=1;j<=n;j++)
      {
         if(board[i]==j)
         {
            printf("row no %d\tcolom no %d\n",i,j);
         }
      }
   }
}
int place(int row, int column)
{
   int i;
   for(i=1;i<=row-1;i++)
            {
               if(board[i]==column)
               {
                  return 0;
                  }
                  else if(abs(board[i]-column)==abs(i-row))
                  {
                     return 0;
                     }
                     }
                     return 1;

         }
         void queen(int row, int n)
         {
            int column;
            for(column=1;column<=n;++column)
            {
               if(place(row,column))
               {
                  board[row]=column;
                  if(row==n)
                  {
                     print(n);
```

```
                    }
                    else
                    {
                       queen(row+1,n);
                       }
                       }
                       }}
```


Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter number of Queens: 4
row no 1   colom no 2
row no 2   colom no 4
row no 3   colom no 1
row no 4   colom no 3
row no 1   colom no 3
row no 2   colom no 1
row no 3   colom no 4
row no 4   colom no 2