```
//Nqueen problem
#include <math.h>
#include <iostream>
using namespace std;

int x[100];
int n;
int count;

int check(int k, int i)
{
   for (int j = 1; j <= k - 1; j++)
   {
      if ((x[j] == i) || (abs(x[j] - i)) == abs(j - k))
         return 0;
   }
   return 1;
}
void nqueen(int k)
{
   for (int i = 1; i <= n; i++)
   {
      if (check(k, i))
      {
         x[k] = i;
         if (k == n)
         {
            count++;
            cout << "Solution " << count << ": ";

            for (int j = 1; j <= n; j++)
            {
               cout << x[j] << " ";
            }
            cout << "\n";
         }
         else
         {
            nqueen(k + 1);
         }
      }
   }
}
int main()
{
   cout << "Enter the number of queens/order of matrix:";
   cin >> n;
   count = 0;

   nqueen(1);

   if (count == 0)
   {
      cout << "No solutions found.\n";
   }
   else
   {
      cout << "Found " << count << " solutions.\n";
   }
 return 0;
}
```

```
PS E:\GIt> cd "e:\GIt\SEM-4\AOA\" ; if ($?) { g++ Nqueen.cpp -o Nqueen } ; if ($?) { .\Nqueen }
Enter the number of queens/order of matrix:4
Solution 1: 2 4 1 3
Solution 2: 3 1 4 2
Found 2 solutions.
PS E:\GIt\SEM-4\AOA> cd "e:\GIt\SEM-4\AOA\" ; if ($?) { g++ Nqueen.cpp -o Nqueen } ; if ($?) { .\Nqueen }
Enter the number of queens/order of matrix:5
Solution 1: 1 3 5 2 4
Solution 2: 1 4 2 5 3
Solution 3: 2 4 1 3 5
Solution 4: 2 5 3 1 4
Solution 5: 3 1 4 2 5
Solution 6: 3 5 2 4 1
Solution 7: 4 1 3 5 2
Solution 8: 4 2 5 3 1
Solution 9: 5 2 4 1 3
Solution 10: 5 3 1 4 2
Found 10 solutions.
PS E:\GIt\SEM-4\AOA>
```