tart here | N-QUEENS USING BACKTRACKING.c

```c
#include<stdio.h>
#include<conio.h>
#include<math.h>
int a[30],count=0;
int place(int pos) {
    int i;
    for (i=1;i<pos;i++) {
        if((a[i]==a[pos])||((abs(a[i]-a[pos])==abs(i-pos))))
           return 0;
    }
    return 1;
}
void print_sol(int n) {
    int i,j;
    count++;
    printf("\n\nSolution #%d:\n",count);
    for (i=1;i<=n;i++) {
        for (j=1;j<=n;j++) {
            if(a[i]==j)
                printf("Q\t"); else
                printf("*\t");
        }
        printf("\n");
    }
}
void queen(int n) {
    int k=1;
    a[k]=0;
    while(k!=0) {
        a[k]=a[k]+1;
        while((a[k]<=n)&&!place(k))
           a[k]++;
        if(a[k]<=n) {
            if(k==n)
                print_sol(n); else {
                k++;
                a[k]=0;
            }
        } else
           k--;
    }
```

art here × N-QUEENS USING BACKTRACKING.c ×

```
11      return 1;
12  }
13  void print_sol(int n) {
14      int i,j;
15      count++;
16      printf("\n\nSolution #%d:\n",count);
17      for (i=1;i<=n;i++) {
18          for (j=1;j<=n;j++) {
19              if(a[i]==j)
20                  printf("Q\t"); else
21                  printf("*\t");
22          }
23          printf("\n");
24      }
25  }
26  void queen(int n) {
27      int k=1;
28      a[k]=0;
29      while(k!=0) {
30          a[k]=a[k]+1;
31          while((a[k]<=n)&&!place(k))
32              a[k]++;
33          if(a[k]<=n) {
34              if(k==n)
35                  print_sol(n); else {
36                  k++;
37                  a[k]=0;
38              }
39          } else
40              k--;
41      }
42  }
43  void main() {
44      int i,n;
45      printf("Enter the number of Queens\n");
46      scanf("%d",&n);
47      queen(n);
48      printf("\nTotal solutions=%d",count);
49      getch();
50  }
51
```