

```

/*****
*Function : place *Description: Function to check whether a queen can
be placed in a specific cell
*Input parameters:*
int x[] : each row of a chess board
int k : position at which the queen has to be placed

RETURNS: true or false
*****/
int place(int x[MAX], int k)
{
int i;
for (i = 1; i < k; i++)
if (x[i] == x[k] || abs(x[i] - x[k]) == abs(i - k))
return 0;
return 1;
}
/*****
*Function : placequeen - Description: Function to display the position
of the queen to be placed
*Input parameters: int n - number of queen
*RETURNS :No return value
*****/
void placequeen(int n)
{
int k, count, x[MAX], i;
k = 1;
count = 0;
x[k] = 0;
printf("\nThe different solutions are as follows");
printf("\n\nEach solution indicates the column in which the Queen is
to be placed in different rows\n");
while (k != 0)
{
x[k] = x[k] + 1;
while ((x[k] <= n) && (!place(x, k)))
x[k] = x[k] + 1;
if (x[k] <= n)
{
if (k == n)
{
count = count + 1;
printf("\n");
for (i = 1; i <= n; i++)
printf("%d\t", x[i]);
printf("\n");
}
else
{
k++;
x[k] = 0;
}
}
}
}

```

```
}  
else  
k--;  
}  
}  
/*****/
```