

|  |
| --- |
| **Ch13-02: *8 Queens*** |
| **Line#** | **Code** |
| 1234567891011121314151617181920212223242526272829303132333435363738394041424344454647484950 | #include <iostream>using namespace std;int Qs[8];int Row = 0;int NoOfAnswers = 0;bool IsValidColumn(int c) { for (int r = 0; r < Row; r++) { //Vertical Check if (Qs[r] == c) return false; //Diagonal check int dR = Row - r; //Alway +ve int dC = c - Qs[r]; //+ve or -ve if (dC < 0) dC = -dC; if (dC == dR) return false; } return true;}void ShowAnswer() { for (int r = 0; r < 8; r++) { for (int c = 0; c < 8; c++) cout << ((Qs[r] == c) ? 'Q' : '.'); cout << endl; } cout << endl;}void Solve() { if (Row < 8) { for (int c = 0; c < 8; c++) { if (IsValidColumn(c)) { Qs[Row] = c; Row++; Solve(); //Recursion Row--; } } } else { NoOfAnswers++; ShowAnswer(); }}int main() { Solve(); cout << "No of Answers is " << NoOfAnswers << endl; return 0;} |

**Sample Answers:**



|  |
| --- |
| Student.h |
| Line# | Code |
| **1****2****3****4****5****6****7****8****9****10****11** | #pragma once**class** Student{**public**: **const char**\* Name; **unsigned char** Age; **float** CGPA; weak\_ptr<Student> LearnFrom; Student(**const char**\* name, **unsigned char** age, **float** cgpa); Student(); ~Student();}; |