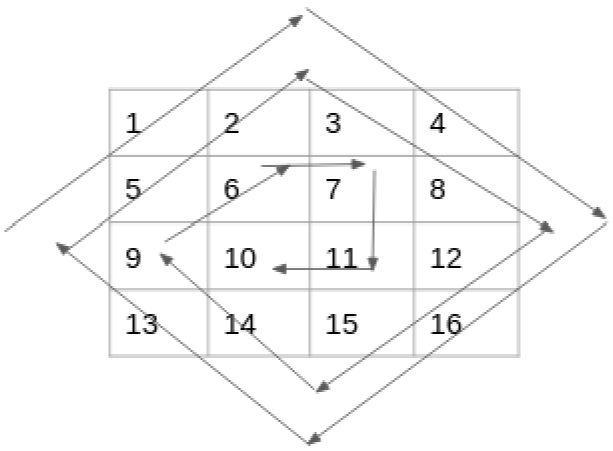
Question:

Traverse a matrix in the given manner



1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Where,

Matrix is of dimension n x n,

n is even

Your function is supposed to return a vector as a solution.

Expected output : **1, 4, 16, 13, 5, 2, 3, 8, 12, 15, 14, 9, 6, 7, 11, 10**

Solution

Expected solution is in O(n), where n is the total number of elements in matrix.

Solution in c++:

```
#include<iostream>
#include<vector>
using namespace std;

vector<int> main(vector<int> &M ){
        int R = M.size() , C = M[0].size();
        int TotalElements = R * C;

        vector<vector<int>> points = {{0, 0}, {0, C - 1}, {R - 1, C - 1}, {R - 1, 0}};
        vector<vector<int>> nextItr = {{1, 0}, {0, -1}, {-1, 0}, {0, 1}};
        vector<vector<int>> shiftInside = {{0, 1}, {1, 0}, {0, -1}, {-1, 0}};
        vector<vector<int>> dir = {{-1, 1}, {1, 1}, {1, -1}, {-1, -1}};

        vector<int> ans;
        int n = 0;
        while(n < TotalElements){
                int fPr = points[0][0], fPc = points[0][1];
            for(int i = 0; i < 4; i++){
                int sr = points[i][0], sc = points[i][1];
                while(sr >= 0 && sr < R && sc >= 0 && sc < C && M[sr][sc] != -1){
                        ans.push_back(M[sr][sc]);
                        n += 1;
                        M[sr][sc] = -1;
                        sr += dir[i][0], sc += dir[i][1];
                   }
                   sr -= dir[i][0], sc -= dir[i][1];
                   if(i == 3){
                        fPr = sr, fPc = sc;
                  }
                  if(n == TotalElements)break;
            }
```

```
          if(n == TotalElements)break;
          else if(fPr - 1 == points[0][0]){
          for(int i = 0; i < 4; i++){
                points[i][0] += shiftInside[i][0];
                points[i][1] += shiftInside[i][1];
          }
        }
        else{
        for(int i = 0; i < 4; i++){
                points[i][0] += nextItr[i][0];
                points[i][1] += nextItr[i][1];
        }
        }

        }

        // Printing traversed matrix
        for(int i = 0; i < ans.size(); i++){
           cout << ans[i] << " ";
        }
        return ans;
}
```