1. Rearrange the array

Difficulty: *Easy*

Question:

Given an unsorted integer array *'arr'*, rearrange it such that

[1st Max, 3rd Max, 5th Max, ... 6th Max, 4th Max, 2nd Max]

Constraint: *Should not sort the array*

Example 1:

Input: arr = [8, 4, 6, 2, 5]

Output: [8, 5, 2, 4, 6]

Example 2:

Input: arr = [1, 2, 3, 4, 5]

Output: [5, 3, 1, 2, 4]

Example 3:

Input: arr= [-1, 2, -3, 9, 7, 9]

Output: [9, 7, -1, -3, 2 , 9]

(Code given below! Try on your own before viewing the code)

Solution : (C++)

```cpp
#include<bits/stdc++.h>
using namespace std;
int main(){

    deque<int> q;

    vector<int> arr {2, 4, -1, 0, -2, 9, 5};

    int n = arr.size(), cnt = 1;

    while(arr.size() > 0){

        // Find the min elem
        auto minelem = min_element(arr.begin(), arr.end());

        // IF array size is odd
        if(n % 2 != 0){
            if(cnt % 2 != 0)
                q.push_front(*minelem);
            else
                q.push_back(*minelem);
        }

        // IF array size is even
        else{
            if(cnt % 2 != 0)
                q.push_back(*minelem);
            else
                q.push_front(*minelem);
        }
        // Remove the min element from the array
        arr.erase(minelem);

        cnt++;
    }

    for(int i=0; i<q.size(); ++i){
        cout<<q[i]<<" ";
    }
}
```