**Reverse Pairs**

#include <bits/stdc++.h>

void mergeArrays(vector<int>& arr, int low, int mid, int high)

{

int left, right;

vector<int> temp;

left = low;

right = mid + 1;

while (left <= mid && right <= high) {

if (arr[left] <= arr[right]) {

temp.push\_back(arr[left]);

left++;

}

else {

temp.push\_back(arr[right]);

right++;

}

}

while (left <= mid) {

temp.push\_back(arr[left]);

left++;

}

while (right <= high) {

temp.push\_back(arr[right]);

right++;

}

for (int i = low; i <= high; i++) {

arr[i] = temp[i - low];

}

}

int countPairs(vector<int>& arr, int low, int mid, int high)

{

int count = 0;

int right = mid + 1;

for (int i = low; i <= mid; i++) {

while (right <= high && arr[i] > 2 \* arr[right]) {

right++;

}

count += right - mid - 1;

}

return count;

}

int mergeSort(vector<int>& arr, int low, int high)

{

int count = 0;

if (low >= high) {

return count;

}

int mid = (low + high) / 2;

count += mergeSort(arr, low, mid);

count += mergeSort(arr, mid + 1, high);

count += countPairs(arr, low, mid, high);

mergeArrays(arr, low, mid, high);

return count;

}

int reversePairs(vector<int>& arr, int n)

{

return mergeSort(arr, 0, n - 1);

}