C++ Assignment Solutions | Merge Sort | Week 12

1. Given an array of integers, sort it in descending order using merge sort algorithm.
Code:

```
#include<bits/stdc++.h>
using namespace std;

void merge(std::vector<int> &a, int low, int mid, int high) {
std::vector<int> b;
int i = low;
int j = mid + 1;
while (i <= mid && j <= high) {
if (a[i] > a[j]) b.push_back(a[i++]);
else b.push_back(a[j++]);
}
while (i <= mid) {
b.push_back(a[i++]);
}
while (j <= high) {
b.push_back(a[j++]);
}
```

2.Reverse Pairs (Leetcode Problem) : Given an integer array nums, return the number of reverse
pairs in the array.
A reverse pair is a pair (i, j) where:
0 <= i < j < nums.length and
nums[i] > 2 * nums[j].

Code:

```
class Solution {
public:
int ans = 0;
void merge(vector<int> &a, int low, int mid, int high) {
int i = low, j = mid+1;
while(i <= mid && j <= high) {
if((long long)a[i] > (long long)2*a[j]) {
ans += mid - i + 1;
j++;
} else {
i++;
}
}
i = low, j = mid+1;
```

```
vector<int> b;
while(i <= mid && j <= high) {
if(a[i] < a[j]) b.push_back(a[i++]);
else b.push_back(a[j++]);
}
```