

Ans 1 - Given two vectors arr1[] and arr2[] of size m and n sorted in increasing order. Merge the two arrays into a single sorted array of size m+n.

```
.vscode > practice.cpp
1  #include <iostream>
2  using namespace std;
3  void mergeArrays(int arr1[], int m, int arr2[], int n, int merged[]) {
4      int i = 0, j = 0, k = 0;
5      while (i < m && j < n) {
6          if (arr1[i] <= arr2[j]) {
7              merged[k++] = arr1[i++];
8          } else {
9              merged[k++] = arr2[j++];
10         }
11     }
12     while (i < m) {
13         merged[k++] = arr1[i++];
14     }
15     while (j < n) {
16         merged[k++] = arr2[j++];
17     }
18 }
19 int main() {
20     int arr1[] = {1, 3, 5, 7};
21     int m = sizeof(arr1)/sizeof(arr1[0]);
22
23     int arr2[] = {2, 4, 6};
24     int n = sizeof(arr2)/sizeof(arr2[0]);
25
26     int merged[m+n];
27     mergeArrays(arr1, m, arr2, n, merged);
28
29     cout << "Merged array: ";
30     for (int i = 0; i < m+n; i++) {
31         cout << merged[i] << " ";
32     }
33     cout << endl;
34     return 0;
```

```
PS D:\cppprogram\vscode> cd "d:\cppprogram\vscode\" ; if ($?) { g++ practice.
ctice }
1100
1001
12
PS D:\cppprogram\vscode> 
```

Ans 2 - Given a vector arr[] sorted in increasing order of n size and an integer x, find if there exists a pair in the array whose sum is exactly x.

```
vscode > arraya.cpp
1  #include <iostream>
2  using namespace std;
3
4  bool findPair(int arr[], int n, int x) {
5      int left = 0, right = n - 1;
6      while (left < right) {
7          int sum = arr[left] + arr[right];
8          if (sum == x) {
9              return true;
10         } else if (sum < x) {
11             left++;
12         } else {
13             right--;
14         }
15     }
16     return false;
17 }
18
19 int main() {
20     int arr[] = {1, 3, 5, 7, 9};
21     int n = sizeof(arr)/sizeof(arr[0]);
22     int x = 12;
23     if (findPair(arr, n, x)) {
24         cout << "Pair with sum " << x << " found.";
25     } else {
26         cout << "Pair with sum " << x << " not found.";
27     }
28     return 0;
29 }
30
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS SQL CONSOLE

```
PS D:\cppprograme\.vscode> cd "d:\cppprograme\.vscode\" ; if ($?) { g++ arraya.c
f ($?) { .\arraya }
Pair with sum 12 found.
PS D:\cppprograme\.vscode>
```


Ans 3 - Given a vector `arr[]` sorted in increasing order of `n` size and an integer `x`, find if there exists a pair in the array whose absolute difference is exactly `x`.

```
.vscode > G+ array2.cpp
2  #include <vector>
3  using namespace std;
4
5  // write a funtion to find square?
6
7  bool findPairWithDiff(vector<int> arr, int x) {
8      int i = 0;
9      int j = 1;
10     int n = arr.size();
11
12     while (i < n && j < n) {
13         if (i != j && arr[j] - arr[i] == x) {
14             return true;
15         }
16         else if (arr[j] - arr[i] < x) {
17             j++;
18         }
19         else {
20             i++;
21         }
22     }
23     return false;
24 }
25 int main() {
26     std::vector<int> arr = {1, 3, 5, 7, 9};
27     int x = 4;
28
29     if (findPairWithDiff(arr, x)) {
30         std::cout << "There exists a pair with absolute difference " << x << std::endl;
31     }
32     else {
33         std::cout << "There does not exist a pair with absolute difference " << x << std::endl;
34     }
35     return 0;
}
```


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS SQL CONSOLE

```
PS D:\cppprograme\.vscode> cd "d:\cppprograme\.vscode\" ; if ($?) { g++
f ($?) { .\array2 }
There exists a pair with absolute difference 4
PS D:\cppprograme\.vscode> █
```

Ans 4 - Given a vector `arr[]` sorted in increasing order. Return an array of squares of each number sorted in increasing order. Where size of vector $1 < \text{size} < 101$.

```
.vscode >  array3.cpp
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  vector<int> sortedSquares(vector<int>& arr) {
6      vector<int> result(arr.size());
7      int left = 0, right = arr.size() - 1;
8      int index = arr.size() - 1;
9
10     while (left <= right) {
11         if (abs(arr[left]) > abs(arr[right])) {
12             result[index--] = arr[left] * arr[left];
13             left++;
14         } else {
15             result[index--] = arr[right] * arr[right];
16             right--;
17         }
18     }
19
20     return result;
21 }
22
23 int main() {
24     vector<int> arr = {-4, -1, 0, 3, 10};
25     vector<int> result = sortedSquares(arr);
26 }
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS  SQL CONS
```

```
PS D:\cppprogramme\.vscode> cd "d:\cppprogramme\.vscode\" ; if ($?) { .\array3 }
0 1 9 16 100
PS D:\cppprogramme\.vscode> 
```

Ans 5 - Given a vector arr[] sorted in increasing order of n size and an integer x, find the number of unique pairs that exist in the array whose absolute sum is exactly x.

```
.vscode > G+ array4.cpp
1  #include <iostream>
2  using namespace std;
3
4  int countPairs(int arr[], int n, int x) {
5      int left = 0, right = n-1, count = 0;
6      while (left < right) {
7          int sum = arr[left] + arr[right];
8          if (sum == x) {
9              count++;
10             // move both pointers to find more pairs
11             left++;
12             right--;
13             while (left < right && arr[left] == arr[left-1]) left++;
14             while (left < right && arr[right] == arr[right+1]) right--;
15         } else if (sum < x) {
16             // move left pointer to increase sum
17             left++;
18         } else {
19             // move right pointer to decrease sum
20             right--;
21         }
22     }
23     return count;
24 }
25 int main() {
26     int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9};
27     int n = sizeof(arr)/sizeof(arr[0]);
28     int x = 7;
29     int count = countPairs(arr, n, x);
30     cout << "Number of pairs with sum " << x << " is " << count << endl;
31     return 0;
32 }
33
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS  SQL C
PS D:\cppprograme\.vscode> cd "d:\cppprograme\.vscode\" ; if
f ($?) { .\array4 }
Number of pairs with sum 7 is 3
PS D:\cppprograme\.vscode>
```

Q6 - Given a vector array nums, print the count of triplets [nums[i], nums[j], nums[k]] such that $i \neq j$, $i \neq k$, and $j \neq k$, and $\text{nums}[i] + \text{nums}[j] + \text{nums}[k] == x$. Where k is an integer given by the user.

```
.vscode > G array5.cpp
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int countTriplets(vector<int>& nums, int k) {
6      int count = 0;
7      int n = nums.size();
8
9      for (int i = 0; i < n - 2; i++) {
10         for (int j = i + 1; j < n - 1; j++) {
11             for (int l = j + 1; l < n; l++) {
12                 if (i != l && i != j && j != l && nums[i] + nums[j] + nums[l] == k) {
13                     count++;
14                 }
15             }
16         }
17     }
18
19     return count;
20 }
21
22 int main() {
23     vector<int> nums = {1, 2, 3, 4, 5};
24     int k = 9;
25
26     int count = countTriplets(nums, k);
27
28     cout << "Number of triplets: " << count << endl;
29
30     return 0;
31 }
32
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS D:\cppprograme\.vscode> cd "d:\cppprograme\"
f ($?) { .\array5 }
Number of triplets: 2
PS D:\cppprograme\.vscode> 
```