

## DAY 5

Maadhav Hira

22BCS10380

KPIT-901(A)

### common element in 3 sorted arrays

```
G: day5ques1.cpp > ...
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  void findCommonElements(const vector<int>& arr1, const vector<int>& arr2, const vector<int>& arr3) {
6      int i = 0, j = 0, k = 0;
7
8      cout << "Common elements: ";
9      while (i < arr1.size() && j < arr2.size() && k < arr3.size()) {
10         if (arr1[i] == arr2[j] && arr2[j] == arr3[k]) {
11             cout << arr1[i] << " ";
12             i++;
13             j++;
14             k++;
15         }
16         else if (arr1[i] < arr2[j]) {
17             i++;
18         }
19         else if (arr2[j] < arr3[k]) {
20             j++;
21         }
22         else {
23             k++;
24         }
25     }
26     cout << endl;
27 }
28
29 int main() {
30     vector<int> arr1 = {1, 5, 10, 20, 40, 80};
31     vector<int> arr2 = {6, 7, 20, 80, 100};
32     vector<int> arr3 = {3, 4, 15, 20, 30, 70, 80, 120};
33
34     findCommonElements(arr1, arr2, arr3);
35
36     return 0;
37 }
38 }
```

Active code page: 65001

C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\" && g++ day5ques1.cpp -o day5ques1 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\"day5ques1  
Common elements: 20 80

c:\Users\Gaurav Kumar\OneDrive\Desktop\class>

## search in 2d matrix

```
day5ques2.cpp > main()
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  bool searchMatrix(const vector<vector<int>>& matrix, int target) {
6      if (matrix.empty() || matrix[0].empty()) {
7          return false;
8      }
9
10     int rows = matrix.size();
11     int cols = matrix[0].size();
12     int left = 0, right = rows * cols - 1;
13
14     // Binary search in the flattened view of the matrix
15     while (left <= right) {
16         int mid = left + (right - left) / 2;
17         int midValue = matrix[mid / cols][mid % cols];
18
19         if (midValue == target) {
20             return true;
21         } else if (midValue < target) {
22             left = mid + 1;
23         } else {
24             right = mid - 1;
25         }
26     }
27     return false;
28 }
29
30
31 int main() {
32     vector<vector<int>> matrix = {
33         {1, 3, 5, 7},
34         {10, 11, 16, 20},
35         {23, 30, 34, 50}
36     };
37     int target = 47;
38
39     if (searchMatrix(matrix, target)) {
40         cout << "Element found!" << endl;
41     } else {
42         cout << "Element not found." << endl;
43     }
44
45     return 0;
46 }
47
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS SEARCH ERROR SPELL CHECKER

Code

Active code page: 65001

```
C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\" && g++ day5ques2.cpp -o day5ques2 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\day5ques2
Element not found.
```

```
c:\Users\Gaurav Kumar\OneDrive\Desktop\class>
```

## Find First and Last Position of Element in Sorted Array.

```
day5ques3.cpp > findPosition(const vector<int>& nums, int, bool)
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int findPosition(const vector<int>& nums, int target, bool findFirst) {
6      int left = 0, right = nums.size() - 1;
7      int result = -1;
8
9      while (left <= right) {
10         int mid = left + (right - left) / 2;
11         if (nums[mid] == target) {
12             result = mid;
13             if (findFirst) {
14                 right = mid - 1;
15             } else {
16                 left = mid + 1;
17             }
18         } else if (nums[mid] < target) {
19             left = mid + 1;
20         } else {
21             right = mid - 1;
22         }
23     }
24     return result;
25 }
26
27 vector<int> findFirstAndLastPosition(const vector<int>& nums, int target) {
28     int firstPos = findPosition(nums, target, true);
29     int lastPos = findPosition(nums, target, false);
30     return {firstPos, lastPos};
31 }
32
33 int main() {
34     vector<int> nums = {5, 7, 7, 8, 8, 10};
35     int target = 8;
36
37     vector<int> result = findFirstAndLastPosition(nums, target);
38     cout << "First Position: " << result[0] << ", Last Position: " << result[1] << endl;
39
40     return 0;
41 }
42
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR SPELL CHECKER

Code

Active code page: 65001

```
C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\" && g++ day5ques3.cpp -o day5ques3 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\day5ques3
First Position: 3, Last Position: 4
```

```
c:\Users\Gaurav Kumar\OneDrive\Desktop\class>
```

## Pair Sum Closest to 0.

```
day5ques4.cpp > closestPairSumToZero(vector<int>& arr)
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  #include <climits>
5  using namespace std;
6
7  pair<int, int> closestPairSumToZero(vector<int>& arr) {
8      if (arr.size() < 2) {
9          return {0, 0};
10     }
11
12     sort(arr.begin(), arr.end());
13
14     int left = 0, right = arr.size() - 1;
15     int closestSum = INT_MAX;
16     pair<int, int> closestPair;
17
18     while (left < right) {
19         int sum = arr[left] + arr[right];
20
21         if (abs(sum) < abs(closestSum)) {
22             closestSum = sum;
23             closestPair = {arr[left], arr[right]};
24         }
25         if (sum < 0) {
26             left++;
27         } else {
28             right--;
29         }
30     }
31
32     return closestPair;
33 }
34
35 int main() {
36     vector<int> arr = {1, 60, -10, 70, -80, 85};
37     pair<int, int> result = closestPairSumToZero(arr);
38
39     cout << "The pair closest to zero sum is: (" << result.first << ", " << result.second << ")" << endl;
40     return 0;
41 }
42
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR SPELL CHECKER
Active code page: 65001
C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\" && g++ day5ques4.cpp -o day5ques4 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\day5ques4
The pair closest to zero sum is: (-80, 85)
c:\Users\Gaurav Kumar\OneDrive\Desktop\class>
```

# Merge k Sorted Lists.

```
day5ques5.cpp > ...
1  #include <iostream>
2  #include <vector>
3  #include <queue>
4  using namespace std;
5
6  struct ListNode {
7      int val;
8      ListNode* next;
9      ListNode(int x) : val(x), next(nullptr) {}
10 };
11
12 struct compare {
13     bool operator()(ListNode* a, ListNode* b) {
14         return a->val > b->val;
15     }
16 };
17
18 ListNode* mergeKLists(vector<ListNode*>& lists) {
19     priority_queue<ListNode*, vector<ListNode*&, compare> minHeap;
20
21     for (auto list : lists) {
22         if (list) {
23             minHeap.push(list);
24         }
25     }
26
27     ListNode* dummy = new ListNode(0);
28     ListNode* tail = dummy;
29
30     while (!minHeap.empty()) {
31         ListNode* smallest = minHeap.top();
32         minHeap.pop();
33
34         tail->next = smallest;
35         tail = tail->next;
36
37         if (smallest->next) {
38             minHeap.push(smallest->next);
39         }
40     }
41
42     ListNode* result = dummy->next;
43     delete dummy;
44     return result;
45 }
46
47 void printList(ListNode* head) {
48     while (head) {
49         cout << head->val << " -> ";
50         head = head->next;
51     }
52     cout << "NULL" << endl;
53 }
54
```

```
55
56 int main() {
57     ListNode* list1 = new ListNode(1);
58     list1->next = new ListNode(4);
59     list1->next->next = new ListNode(5);
60
61     ListNode* list2 = new ListNode(1);
62     list2->next = new ListNode(3);
63     list2->next->next = new ListNode(4);
64
65     ListNode* list3 = new ListNode(2);
66     list3->next = new ListNode(6);
67
68     vector<ListNode*> lists = {list1, list2, list3};
69
70     ListNode* result = mergeKLists(lists);
71
72     cout << "Merged list: ";
73     printList(result);
74
75     return 0;
76 }
77
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS SEARCH ERROR SPELL CHECKER

Active code page: 65001

```
C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\" && g++ day5ques5.cpp -o day5ques5 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\"day5ques5
Merged list: 1 -> 1 -> 2 -> 3 -> 4 -> 4 -> 5 -> 6 -> NULL
c:\Users\Gaurav Kumar\OneDrive\Desktop\class>
```

## median of two sorted arrays.

```
day5ques6.cpp > ...
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  #include <limits>
5  using namespace std;
6
7  double findMedianSortedArrays(const vector<int>& nums1, const vector<int>& nums2) {
8      vector<int> merged(nums1.size() + nums2.size());
9      merge(nums1.begin(), nums1.end(), nums2.begin(), nums2.end(), merged.begin());
10
11      int n = merged.size();
12      if (n % 2 == 0) {
13          return (merged[n / 2 - 1] + merged[n / 2]) / 2.0;
14      } else {
15          return merged[n / 2];
16      }
17  }
18
19  int main() {
20      int n1, n2;
21      cout << "Enter the number of elements in the first array: ";
22      cin >> n1;
23
24      vector<int> nums1(n1);
25      cout << "Enter the elements of the first sorted array: ";
26      for (int i = 0; i < n1; i++) {
27          cin >> nums1[i];
28      }
29
30      cout << "Enter the number of elements in the second array: ";
31      cin >> n2;
32
33      vector<int> nums2(n2);
34      cout << "Enter the elements of the second sorted array: ";
35      for (int i = 0; i < n2; i++) {
36          cin >> nums2[i];
37      }
38
39      double median = findMedianSortedArrays(nums1, nums2);
40      cout << "Median: " << median << endl;
41
42      return 0;
43  }
44  }
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR  SPELL CHECKER  Code
Active code page: 65001
C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class" && g++ day5ques6.cpp -o day5ques6 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\day5ques6
Enter the number of elements in the first array: 2
Enter the elements of the first sorted array: 1 2
Enter the number of elements in the second array: 1
Enter the elements of the second sorted array: 3
Median: 2
c:\Users\Gaurav Kumar\OneDrive\Desktop\class>
```

## searching a number

```
day5ques7.cpp > ...
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 int main() {
6     int n, target;
7
8     cout << "Enter the number of elements in the array: ";
9     cin >> n;
10
11     vector<int> arr(n);
12     cout << "Enter the elements of the array: ";
13     for (int i = 0; i < n; i++) {
14         cin >> arr[i];
15     }
16
17     cout << "Enter the number to search: ";
18     cin >> target;
19
20     bool found = false;
21     for (int i = 0; i < n; i++) {
22         if (arr[i] == target) {
23             cout << "Number " << target << " found at position " << i + 1 << ".\n" << endl;
24             found = true;
25             break;
26         }
27     }
28
29     if (!found) {
30         cout << "Number " << target << " not found in the array.\n" << endl;
31     }
32
33     return 0;
34 }
35
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS SEARCH ERROR SPELL CHECKER

Active code page: 65001

```
C:\Users\Gaurav Kumar\OneDrive\Desktop\class>cd "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\" && g++ day5ques7.cpp -o day5ques7 && "c:\Users\Gaurav Kumar\OneDrive\Desktop\class\"day5ques7
Enter the number of elements in the array: 5
Enter the elements of the array: 1 2 3 4 5
Enter the number to search: 3
Number 3 found at position 3.

c:\Users\Gaurav Kumar\OneDrive\Desktop\class>
```