

# DSA Lab 1 - Arrays (53457) M. Abdullah

## Problem 1:

Write a function that takes an array of integers and returns the sum of its elements.

## Problem Solving

Lab Task 1:  
Sum of Elements:

Algo's      double      sum (~~A[]~~, size)

```
sum = 0
for (i=0; i<size; i++)
    sum = sum + A[i]
return sum;
```

Dab

## Code

```

#include<iostream>
using namespace std;
class Lab2 {
public:
    double prob1(int arr[],int size) {
        double sum=0;
        for(int i=0;i<size;i++)
        {
            sum=sum+arr[i];
        }
        return sum;
    }
};

int main() {
    Lab2 obj;
    int arr[5]={1,2,3,4,5};
    double sum = obj.prob1(arr,5);
    cout<<sum;
    return 0;
}

```

## Output

The screenshot shows the Visual Studio Code interface. The code editor displays the C++ code provided above. The terminal window at the bottom shows the command line output of the program's execution. The file explorer on the left shows the project structure and files.

```

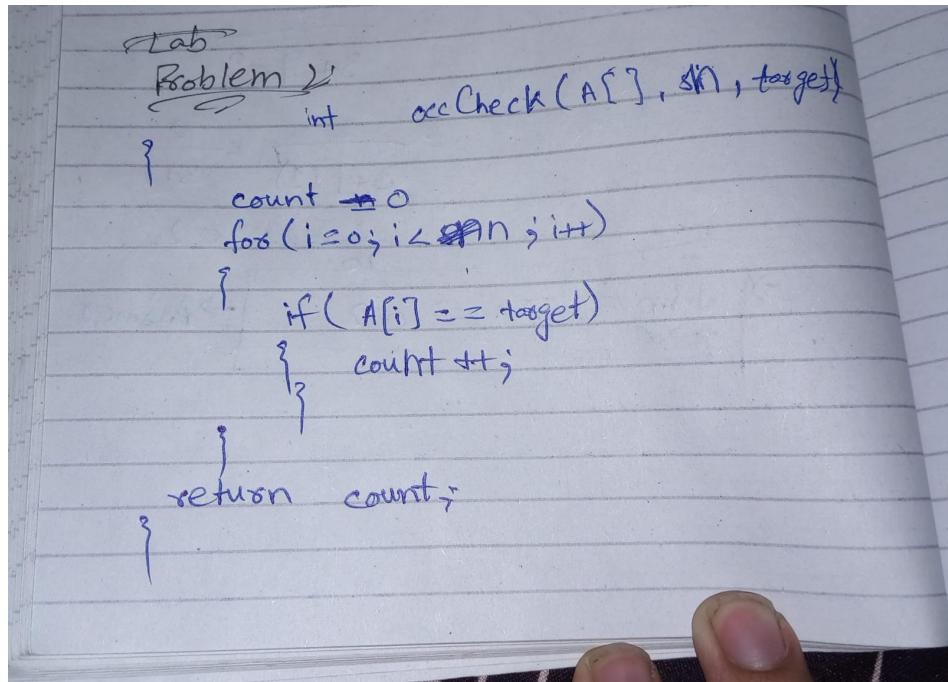
PS D:\Resources> & "c:/Users/hp/.vscode/extensions/ms-vscode.cpptools-1.22.9-win32-x64/debugAdapters/bin/WindowsDebugLauncher.exe" "--stdIn=Microsoft-MIEngine-In-nkuyos4b.pek" "--stdout=Microsoft-MIEngine-Out-23rc4sat.qvk" "--stderr=Microsoft-MIEngine-Error-zu10ig1x.tcc" "--pid=Microsoft-MIEngine-Pid-35z5bicq.3ad" "--dbgExe=C:\nys64\curl64\bin\gdb.exe" "--interpreter=mi"
15
PS D:\Resources>

```

## Problem 2:

Given an array and a target element, count the occurrences of the target element in the array.

## Problem Solving



## Code

```
#include<iostream>
using namespace std;
class Lab2 {
public:
    int prob2(int arr[], int target) {
        int size = sizeof(arr);
        int count = 0;
        for(int i=0; i<size; i++) {
            if(target==arr[i])
            {
```

```

        count++;
    }
}
return count;
}
};

int main() {
    Lab2 obj;
    int arr[8]={19,20,32,1123,238,899,89,20};
    int target = 20;
    cout<<target<<" is found "<<obj.prob2(arr,target)<<" times.";
    return 0;
}

```

## Output

The screenshot shows the Microsoft Visual Studio Code interface. The code editor displays the provided C++ code. The terminal window at the bottom shows the output of running the program, which includes the command to start the debugger and the result "20 is found 2 times." The file explorer on the left shows a project structure with multiple files and folders related to a DSA lab assignment.

```

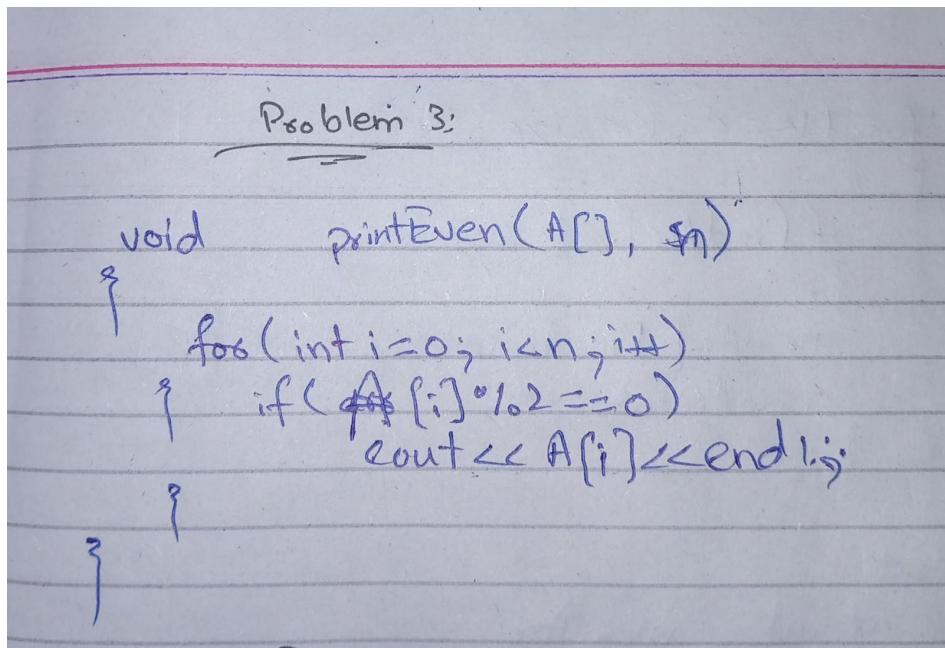
PS D:\Resources> & 'c:\Users\hp\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-ajmd1cxm2' '--stdout=Microsoft-MIEngine-Out-1gbvitjq.qht' '--stderr=Microsoft-MIEngine-Error-izaqobtc.3ic' '--pid=Microsoft-MIEngine-Pid-vkdsmkr.yri' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
20 is found 2 times.
PS D:\Resources>

```

## Problem 3:

Write a function that returns all the even numbers from an array.

# Problem Solving



## Code

```
#include<iostream>
using namespace std;
class Lab2 {
public:
void prob3(int arr[]) {

    int size = sizeof(arr);
    for(int i=0;i<size;i++)
    {
        if(arr[i]%2==0)
        {
            cout<<arr[i]<<endl;
        }
    }
};

int main() {
```

```

Lab2 obj;
int arr[8]={19, 20, 32, 1123, 238, 899, 18, 93};
obj.prob3(arr);
return 0;
}

```

## Output

The screenshot shows the Microsoft Visual Studio Code interface. The code editor displays a C++ file named `Problem_3.cpp` with the following content:

```

1 #include<iostream>
2 using namespace std;
3 class Lab2 {
4 public:
5     void prob(int arr[]) {
6         int n = sizeof(arr)/sizeof(arr[0]);
7         sort(arr, arr+n);
8         for (int i=0; i<n; i++)
9             cout << arr[i] << " ";
10    }
11 }

```

The terminal window shows the command line output of the program's execution:

```

PS D:\Resources> & 'c:\Users\hp\vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapte
rs\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-p24q0fw5.oci' '--stdout=Microsoft-M
IEngine-Out-sco4k5ux.okq' '--stderr=Microsoft-MIEngine-Error-nuuh2hp2.teq' '--pid=Microsoft-MIEngi
n-e-Pid-1f1ds3vg.dsu' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
20
32
238
18
PS D:\Resources>

```

The file explorer sidebar shows a project structure under `RESOURCES`, including files like `Problem_1.cpp`, `Problem_1.exe`, and `Problem_3.cpp`. The status bar at the bottom indicates the current file is `main.cpp`.

## Problem 4:

Given an array of integers, remove duplicates from the array and return the new array.

## Problem Solving

### Problem B1:

Deleting Duplicate

void( int arr[], size)

Start:

int Arr[ size ]

in int Arr[ i for all index ] = 0

duplicateCount = 0

for ( i = size - 1; i >= 0; i -- ) {

{ for ( j = i - 1; j >= 0; j -- )

{ if ( Arr[i] = arr[j] )

int Arr[i] = 1;

duplicateCount ++ )

}

}

newArr[ size - duplicateCount ]

newArrIndex = 0

```

for (k=0; k<size; k++)
{
    if (indArr[k] == 0)
    {
        newArr[newArrIndex] = arr[k];
        newArrIndex++;
    }
}
cout
Output newArr
for (i=0; i<(size-duplicateCount); i++)
{
    cout << newArr[i];
}

```

## Code

```

#include<iostream>
using namespace std;
void prob4(int arr[], int size)
{
    //bool isdup = false;

```

```

int indArr[size];
for(int i=0;i<size;i++)
{
    indArr[i] =0;
}
int dupCount=0;
for(int i=size-1;i>=0;i--)
{
    for(int j=i-1;j>=0;j--)
    {
        if(arr[i]==arr[j])
        {
            indArr[i] = 1;
            dupCount++;
        }
    }
}
int newArr[size-dupCount];
int newArrIndex=0;
for(int k=0;k<size;k++)
{
    if(indArr[k]==0)
    {
        newArr[newArrIndex] = arr[k];
        newArrIndex++;
    }
}
cout<<"New Array is: ";
for(int i=0;i<(size-dupCount);i++)
{
    cout<<newArr[i]<<endl;
}
}

int main() {
    int size = 9;

```

```

int arr[size];
cout<<"Enter Elements of Array with duplicates: ";
for(int i=0;i<size;i++)
{
    cin>>arr[i];
}
prob4(arr,size);
return 0;
}

```

## Output

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder structure under "RESOURCES" containing "University", "3rd Sem", "DSA Labs", and "Lab 2-08.Oct.2024". Inside "Lab 2-08.Oct.2024", there are multiple files: Problem\_1.cpp, Problem\_1.exe, Problem\_2.cpp, Problem\_2.exe, Problem\_3.cpp, Problem\_3.exe, Problem\_4.cpp, Problem\_4.exe, Problem\_5.cpp, Problem\_6.cpp, Problem\_6.exe, Problem\_7.cpp, Problem\_8.cpp, Problem\_8.exe, Problem\_9.cpp, Problem\_10.cpp, and Problem\_11.cpp.
- Terminal:** Displays the command-line output of the program execution. It shows the user entering elements of the array and the resulting sorted array.
- Status Bar:** Shows the current file is "main.cpp", the line number is 1, the column number is 111, and the file size is 0. It also displays system information like battery level (28%), temperature (28°C), and date/time (10:18 pm, 18/10/2024).

```

PS D:\Resources> & 'c:\Users\hp\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-5cbduueh.cmz' '--stdout=Microsoft-MIEngine-Out-3gt44ba2.qhb' '--stderr=Microsoft-MIEngine-Error-lnkpkcmx.vop' '--pid=Microsoft-MIEngine-Pid-bq05dtb.jih' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter Elements of Array with duplicates: 172
1
32
76
172
New Array is: 172
1
32
76
11
2
87
111
1
2
87
111
PS D:\Resources> & 'c:\Users\hp\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-p24q0fv5.oc1' '--stdout=Microsoft-MIEngine-Out-sco4k5ux.ok' '--stderr=Microsoft-MIEngine-Error-nuh2hp2.teq' '--pid=Microsoft-MIEngine-Pid-1f1ds3vg.ds1' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'

```

## Problem 5:

Write a function to check if an array is sorted in ascending order.

## Problem Solving

## Problem 5:

isSorted

bool isSorted( A[], n )

{  
    for ( i = 0; i < n - 1; i++ )

        {  
            for ( j = i + 1; j < n; j++ )

                {  
                    if ( ~~A[i]~~ > A[j] )

                        return false;  
                }

    }

}

    return true;

}

## Code

```
#include<iostream>
using namespace std;
```

```

bool isSorted(int arr[],int size)
{
    for(int i=0;i<size-1;i++)
    {
        for(int j=i+1;j<size;j++)
        {
            if(arr[i]>arr[j])
            {
                return false;
            }
        }
    }
    return true;
}
int main()
{
    int size = 10;
    int arr[size];
    for(int i=0;i<size;i++)
    {
        cin>>arr[i];
    }
    if(isSorted(arr,size))
    {
        cout<<"Array is Sorted";
    }
    else
    {
        cout<<"Array is not Sorted";
    }
    return 0;
}

```

## Output

```
bool isSorted(int arr[], int size)
{
    for(int i=0; i<size-1; i++)
    {
        if(arr[i] > arr[i+1])
            return false;
    }
    return true;
}
```

## Problem 6:

Rotate an array to the left by a given number of positions.

## Problem Solving

: Problem 6:

```

void rotateArr(int num)
{
    num = num % size ( if number
                        is bigger than
                        array size)
    *temp arr = new int[size];
    for loop (i < size)
    {
        temp[i] = arr[(i + num) % size];
    }
    for (i < size)
    {
        arr[i] = temp[i];
    }
}

```

## Code

```

#include <iostream>
using namespace std;

class Lab2 {
private:
    int* arr;
    int size;

```

```
public:

    Lab2(int s) {
        size = s;
        arr = new int[size];
    }

    void inputArray() {
        cout << "Enter the elements of the array: " << endl;
        for (int i = 0; i < size; i++) {
            cin >> arr[i];
        }
    }

    void rotateArray(int num) {
        num = num % size;
        int* temp = new int[size];

        for (int i = 0; i < size; i++) {
            temp[i] = arr[(i + num) % size];
        }

        for (int i = 0; i < size; i++) {
            arr[i] = temp[i];
        }

        delete[] temp;
    }

    // Function to display the array
    void displayArray() const {
        cout << "Array after rotation: " << endl;
        for (int i = 0; i < size; i++) {
```

```
        cout << arr[i] << " ";
    }
    cout << endl;
}

int main() {
    int size;
    cout << "Enter the size of the array: ";
    cin >> size;

    Lab2 l2(size);

    l2.inputArray();

    int num;
    cout << "Enter the number of rotations to perform: ";
    cin >> num;

    l2.rotateArray(num);
    l2.displayArray();

    return 0;
}
```

## Output

```
PS D:\Resources> & 'c:\Users\hp.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-3k1mq1g.ulr' '--stdout=Microsoft-MIEngine-Out-vtmofblu.rft' '--stderr=Microsoft-MIEngine-Error-0ekfzs2m.ti3' '--pid=Microsoft-MIEngine-Pid-anc2kht.g5v' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter the size of the array: 4
Enter the elements of the array:
12
433
11
2
Enter the number of rotations to perform: 3
Array after rotation:
2 12 433 11
PS D:\Resources>
```

## Problem 7:

Write a function to find the second largest element in an array.

## Problem Solving

## Problem 7:

2nd Largest in Array

- \* Sort in Ascending order
- \* Element at  $\rightarrow 2$  index  
or size-2 index

## Code

```
#include <iostream>
using namespace std;

class Lab2 {
private:
    int* arr;
    int size;

public:
    Lab2(int s) {
        size = s;
        arr = new int[size];
    }
}
```

```

void inputArray() {
    cout << "Enter the elements of the array: " << endl;
    for (int i = 0; i < size; i++) {
        cin >> arr[i];
    }
}

void sortArray() {
    for (int i = 0; i < size - 1; i++) {
        for (int j = 0; j < size - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}

// Function to find the second largest element
int findSecondLargest() {
    // Since the array is sorted, the second largest element
    // will be the second last element
    if (size < 2) {
        cout << "Array must have at least two elements to
find the second largest." << endl;
        return -1;
    }
    return arr[size - 2];
}

};


```

```

int main() {
    int size;
    cout << "Enter the size of the array: ";
    cin >> size;

    Lab2 l2(size);

    l2.inputArray();

    l2.sortArray();

    int secondLargest = l2.findSecondLargest();
    cout << "The second largest element is: " << secondLargest
        << endl;

    return 0;
}

```

## Output

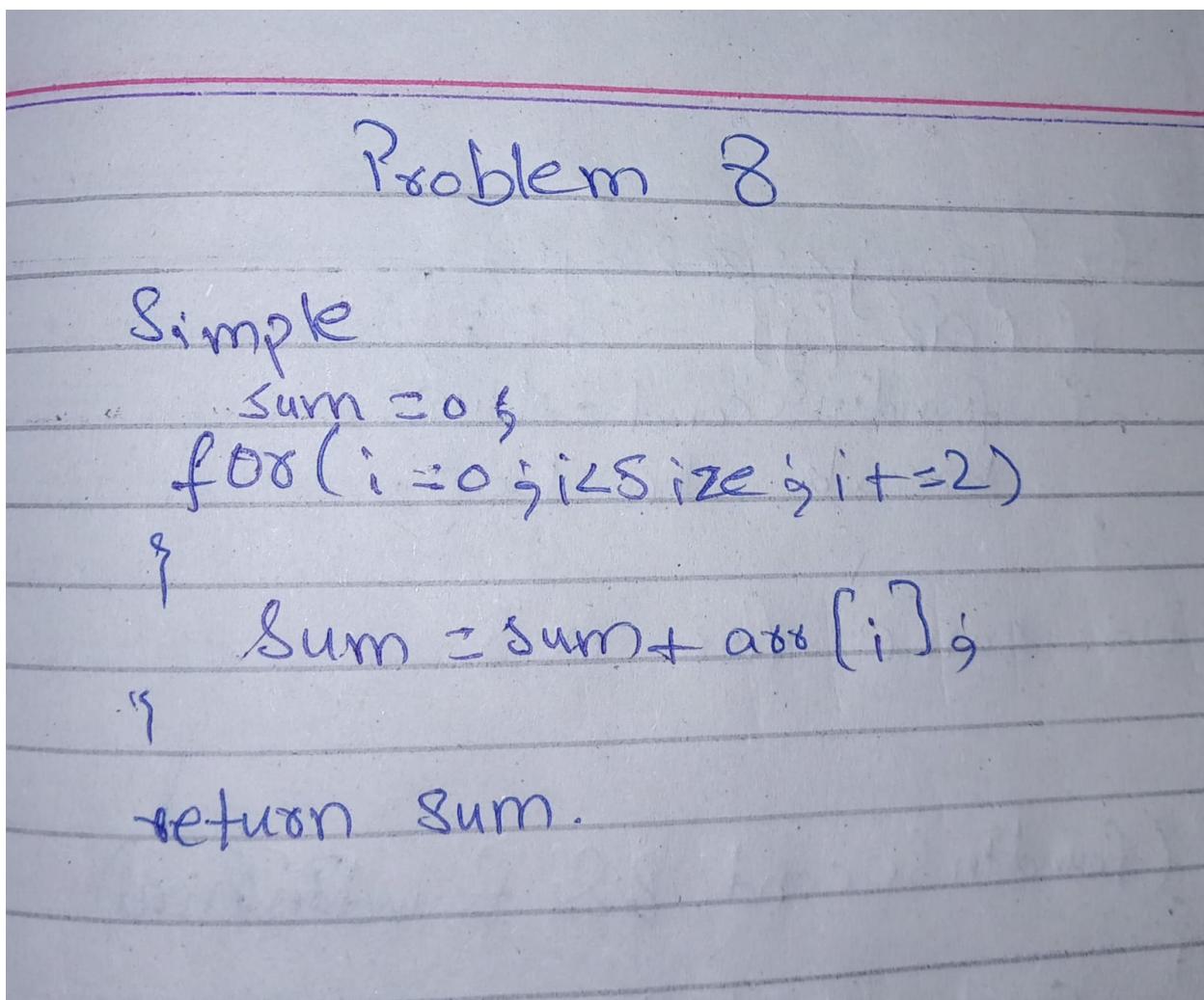
The screenshot shows the Microsoft Visual Studio Code interface with the following details:

- File Explorer:** Shows a project structure under 'RESOURCES' with folders for 'University', '3rd Sem', 'DSA', and 'Labs'. Inside 'Labs' is a folder 'Lab 2-08.Oct.2024' containing files 'Problem\_1.cpp' through 'Problem\_10.cpp'.
- Terminal:** The active tab is 'C++ Problem\_7.cpp U'. It shows the command being run: 'PS D:\Resources> & "c:\Users\vp\vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debug\adpt\rslab\bin\WindowsDebugLauncher.exe" "-std=noneMicrosoft-MEngine-In-FailSafe;qrm" "--stdout=Microsoft-MEngine-Out-fu5j1xi.apm" "-std=noneMicrosoft-MEngine-Error-hcumbjjj3rv" "-pid=Microsoft-MEngine-e-Plat-n12wbnr\_rca" "--dpeFwec:\mysys64\ucrt64\bin\gdb.exe" '--interpreter=mi'.
- Output:** The terminal also displays the user's input: 'Enter the size of the array: 5' followed by the array elements '12', '312', '1', '23', and '21'. The program then outputs 'The second largest element is: 23'.
- Status Bar:** Shows the current file is 'main.cpp', line 66, column 1, 1419 selected, spaces: 4, UTF-8, CRLF, C++, AI Code Chat, Win32, and the date/time as 18/10/2024 10:22 pm.

## Problem 8:

: Find the sum of all elements at even indices in an array

## Problem Solving



## Code

```
#include<iostream>
using namespace std;
int prob8(int arr[], int size)
{
    int sum = 0;
```

```

for(int i=0;i<size;i+=2)
{
    sum = sum+arr[i];
}
return sum;
}

int main()
{
    int size = 6;
    int arr[size];
    cout<<"Enter elements of Array: ";
    for(int i=0;i<size;i++)
    {
        cin>>arr[i];
    }
    cout<<"Sum of elements at even indices = "<<prob8(arr,size);
    return 0;
}

```

## Output

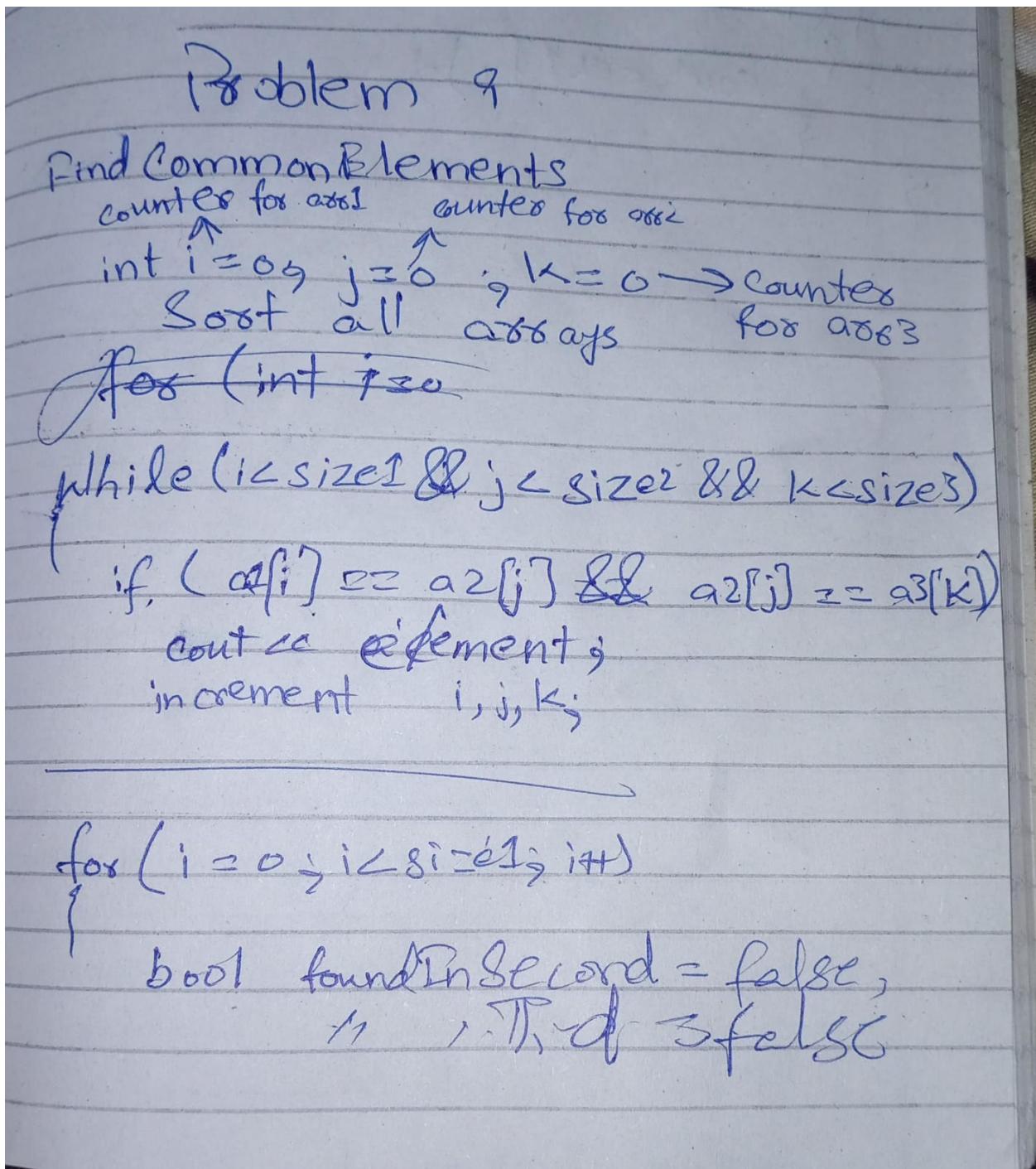
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project structure under "University > 3rd Sem > DSA > Labs > Lab 2-08.Oct.2024".
- Terminal:** Displays the C++ code and its execution. The code includes a function to calculate the sum of even indices of an array. The terminal shows the user inputting array elements (53, 12, 32, 12, 44, 11) and the resulting output: "Sum of elements at even indices = 129".
- Status Bar:** Shows the current file is "main.cpp", line 23, column 2, and the date/time as 18/10/2024 10:23 pm.

## Problem 9:

Given three sorted arrays, find the common elements in all three.

## Problem Solving



```
for (j=0; j<size2; j++)  
    if (arr2[j] == arr1[i])  
        foundInSecond = true;  
        break the loop;
```

}

Do same for a3

```
if (foundInSecond && foundInThird)  
    cout << arr1[i];
```

}

## Code

```
#include <iostream>  
using namespace std;  
  
class Lab2 {  
private:
```

```
int* arr1;
int* arr2;
int* arr3;
int size1, size2, size3;

public:
    Lab2(int s1, int s2, int s3) {
        size1 = s1;
        size2 = s2;
        size3 = s3;
        arr1 = new int[size1];
        arr2 = new int[size2];
        arr3 = new int[size3];
    }

    void inputArrays() {
        cout << "Enter the elements of the first array: " <<
endl;
        for (int i = 0; i < size1; i++) {
            cin >> arr1[i];
        }

        cout << "Enter the elements of the second array : " <=
< endl;
        for (int i = 0; i < size2; i++) {
            cin >> arr2[i];
        }

        cout << "Enter the elements of the third array: " <<
endl;
        for (int i = 0; i < size3; i++) {
            cin >> arr3[i];
        }
    }

    void findCommonElements() {
```

```

        cout << "Common elements are: ";

        for (int i = 0; i < size1; i++) {
            bool foundInSecond = false, foundInThird = false;

            for (int j = 0; j < size2; j++) {
                if (arr1[i] == arr2[j]) {
                    foundInSecond = true;
                    break;
                }
            }

            for (int k = 0; k < size3; k++) {
                if (arr1[i] == arr3[k]) {
                    foundInThird = true;
                    break;
                }
            }

            if (foundInSecond && foundInThird) {
                cout << arr1[i] << " ";
            }
        }

        cout << endl;
    }
};

int main() {
    int size1, size2, size3;

    cout << "Enter the size of the first array: ";
    cin >> size1;

    cout << "Enter the size of the second array: ";
    cin >> size2;
}

```

```

cout << "Enter the size of the third array: ";
cin >> size3;

Lab2 l2(size1, size2, size3);

l2.inputArrays();

l2.findCommonElements();

return 0;
}

```

## Output

```

PS D:\Resources> & "c:/Users/vhp/.vscode/extensions/ms-vscode.cpptools-1.22.9-win32-x64/debugAdapters/bin/WindowsDebugLauncher.exe" "--stdIn=Microsoft-MIEngine-In-to4usq,tba" "--stdout=Microsoft-MIEngine-Out-addfsql,qsl" "--stderr=Microsoft-MIEngine-Error-uoksb4,eq1" "--pid=Microsoft-MIEngine-ProcessId-1025" "c:/Users/vhp/.vscode/extensions/ms-vscode.cpptools-1.22.9-win32-x64/bin/gdb.exe" "--interpreter=mi"
Enter the size of the first array: 3
Enter the size of the second array: 4
Enter the size of the third array: 5
Enter the elements of the first array:
12
3
112
Enter the elements of the second array:
3
123
123
5
Enter the elements of the third array:
5
12
3
123
44
Common elements are:
PS D:\Resources>

```

## Problem 10:

Given an array of  $n+1$  integers where each integer is between 1 and  $n$ , there is exactly one duplicate. Find the duplicate number.

## Problem Solving

### Problem 10:

We have  $n+1$  Array  
where elements =  $\{1-n\}$  and

duplicate element is 1

Let  $n = 6$       size =  $7 = 6+1 = n+1$

$\Rightarrow A[6+1] = \{1|2|3|4|5|6|5\}$

int duplicate(A[], num) {  
 for (i=0; i<n+1; i++)  
 for (j=0; j<n+1; j++)  
 if (A[i] == A[j])  
 return A[i];  
}

## Code

```
#include <iostream>
using namespace std;

class Lab2 {
private:
    int* arr;
    int size;

public:
```

```

Lab2(int s) {
    size = s;
    arr = new int[size];
}

void inputArray() {
    cout << "Enter the elements of the array (between 1 and " << size - 1 << "): " << endl;
    for (int i = 0; i < size; i++) {
        cin >> arr[i];
    }
}

int findDuplicate() {
    for (int i = 0; i < size; i++) {
        for (int j = i + 1; j < size; j++) {
            if (arr[i] == arr[j]) {
                return arr[i];
            }
        }
    }
    return -1; // In case no duplicate is found
}

void displayArray() const {
    cout << "Array elements: ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
}

int main() {
    int size;
    cout << "Enter the size of the array (n + 1): ";

```

```

    cin >> size;

    Lab2 lab2(size);

    lab2.inputArray();

    int duplicate = lab2.findDuplicate();
    if (duplicate != -1) {
        cout << "The duplicate element is: " << duplicate <<
endl;
    } else {
        cout << "No duplicate element found." << endl;
    }

    return 0;
}

```

## Output

The screenshot shows a Windows desktop environment with Visual Studio Code open. The code editor displays a C++ file named `Problem_10.cpp`. The terminal window shows the following interaction:

```

PS D:\Resources> & "c:\Users\hp\vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters\WindowsDebugLauncher.exe" "--std=msvc2019" "--target=Microsoft-MIEngine-Error-pauseitip.10f" "--pid=Microsoft-MIEngine-Error-pauseitip.10f" "--interpreter=mimic"
Enter the size of the array (n + 1): 6
Enter the elements of the array (between 1 and 5):
1
2
3
4
5
2
The duplicate element is: 2
PS D:\Resources>

```

The terminal also shows multiple instances of the `cppdbg` process running, corresponding to the multiple breakpoints set in the code.