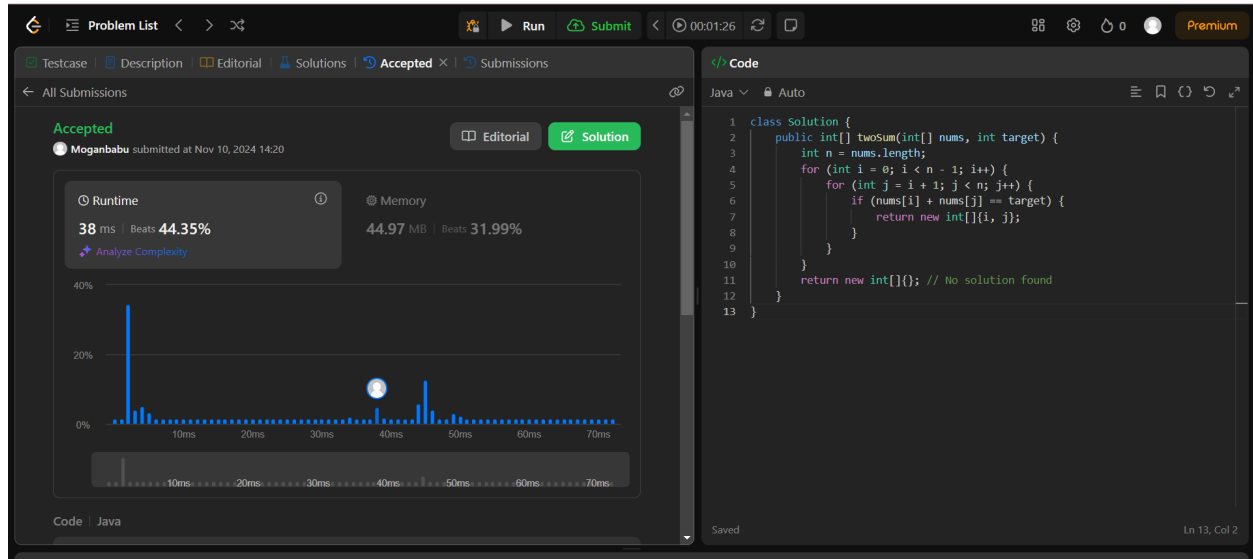


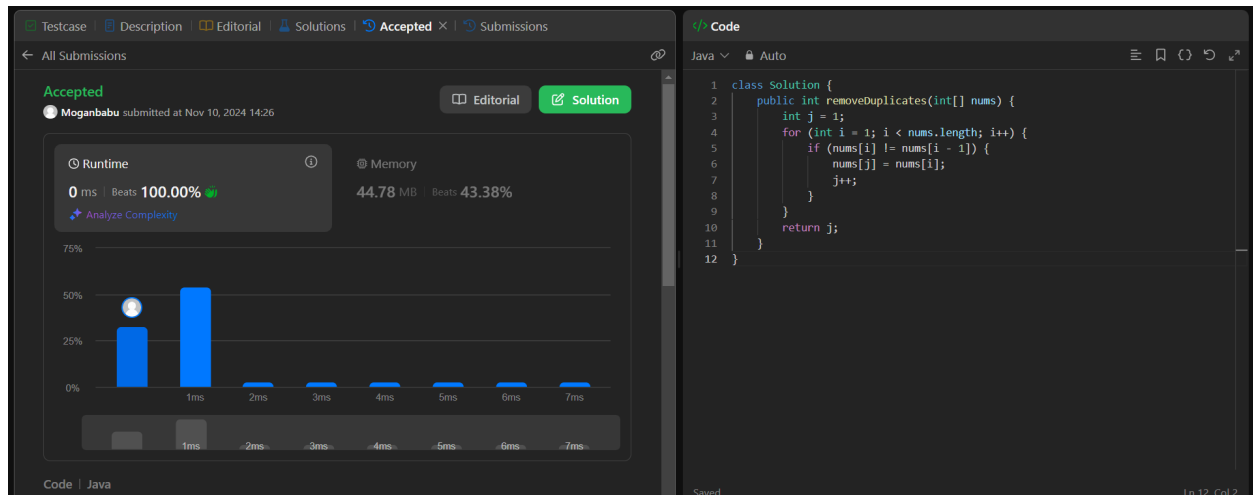
MOGAN BABU C

22CZ031

1. Two Sum



2. Leetcode 26



3. Leetcode 27

Accepted

Moganbabu submitted at Nov 10, 2024 14:28

Runtime: 0 ms | Beats 100.00%
Memory: 42.03 MB | Beats 19.41%

Code | Java

```
1 class Solution {
2     public int removeElement(int[] nums, int val) {
3         int k = 0;
4
5         for (int i = 0; i < nums.length; i++) {
6             if (nums[i] != val) {
7                 nums[k] = nums[i];
8                 k++;
9             }
10        }
11
12        return k;
13    }
14 }
```

4. Leetcode 1480

Accepted

Moganbabu submitted at Nov 10, 2024 14:30

Runtime: 0 ms | Beats 100.00%
Memory: 42.49 MB | Beats 76.36%

Code | Java

```
1 class Solution {
2     public int[] runningSum(int[] nums) {
3
4         for (int i = 1; i < nums.length; i++) {
5             nums[i] = nums[i-1] + nums[i];
6         }
7
8         return nums;
9     }
10 }
```

Test Result

5. Leetcode 509

The screenshot shows the LeetCode interface for problem 509. The 'Accepted' status is confirmed, with a submission by 'Moganbabu' from Nov 10, 2024. The runtime is 7 ms, beating 47.38% of submissions, and the memory usage is 40.78 MB, beating 6.98%. A bar chart shows the distribution of runtime times, with the majority of submissions falling between 7ms and 8ms. The code is a Java solution for calculating the nth Fibonacci number using a recursive approach.

```
class Solution {
    static int fib(int n) {
        if (n <= 1) {
            return n;
        }
        return fib(n-1) + fib(n-2);
    }

    public static void main (String[] args) {
        int a = 10;
        for (int i = 0; i <= 10; i++) {
            System.out.print(fib(i));
        }
    }
}
```

6. selection sort

The screenshot shows a Java IDE with a project named 'javaprogs'. The code implements a selection sort algorithm. The 'Main.java' file contains the following code:

```
import java.util.Arrays;

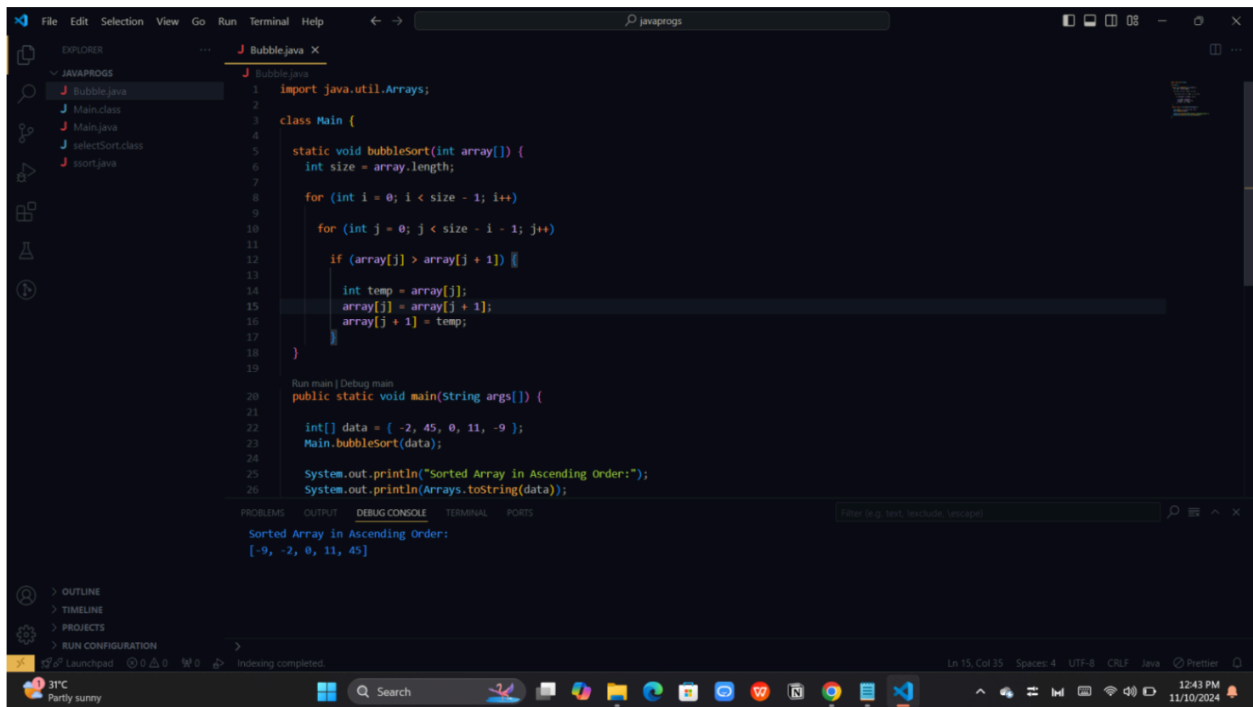
class selectSort {
    void selectionSort(int[] arr) {
        int n = arr.length;
        for (int i = 0; i < n - 1; i++) {
            int min = i;
            for (int j = i + 1; j < n; j++) {
                if (arr[j] < arr[min]) {
                    min = j;
                }
            }
            int temp = arr[i];
            arr[i] = arr[min];
            arr[min] = temp;
        }
    }

    public static void main (String args[]) {
        int[] data = { 20, 12, 10, 15, 2 };
        selectSort ss = new selectSort();
        ss.selectionSort(data);
        System.out.println("Sorted Array in Ascending Order: ");
        System.out.println(Arrays.toString(data));
    }
}
```

The terminal output shows the execution of the program:

```
PS C:\Users\prasa\OneDrive\Documents\javaprogs> javac ssort.java
PS C:\Users\prasa\OneDrive\Documents\javaprogs> java ssort
Error: Could not find or load main class ssort
Caused by: java.lang.ClassNotFoundException: ssort
PS C:\Users\prasa\OneDrive\Documents\javaprogs> java ssort.java
Sorted Array in Ascending Order:
[2, 10, 12, 15, 20]
PS C:\Users\prasa\OneDrive\Documents\javaprogs>
```

7. Bubble sort

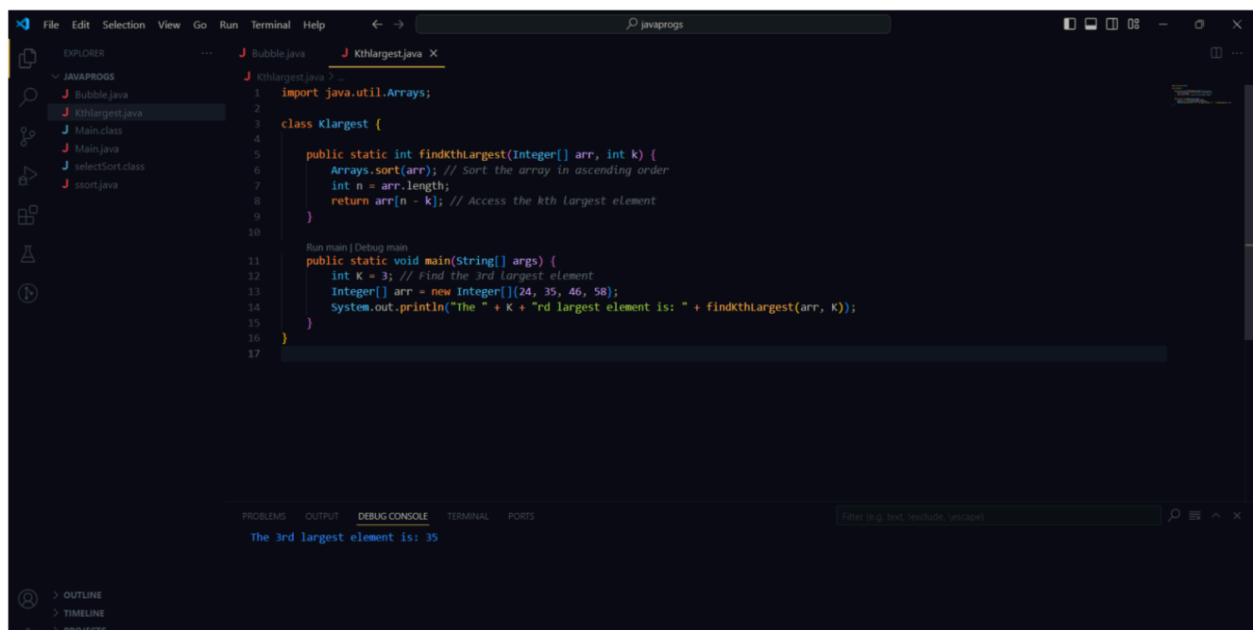


The screenshot shows an IDE with a Java project named 'javaprogs'. The file explorer on the left shows a package 'JAVAPROGS' containing files 'Bubble.java', 'Main.class', 'Main.java', 'selectSort.class', and 'sort.java'. The 'Bubble.java' file is open in the editor, showing the following code:

```
1 import java.util.Arrays;
2
3 class Main {
4
5     static void bubbleSort(int array[]) {
6         int size = array.length;
7
8         for (int i = 0; i < size - 1; i++)
9
10            for (int j = 0; j < size - i - 1; j++)
11
12                if (array[j] > array[j + 1]) {
13
14                    int temp = array[j];
15                    array[j] = array[j + 1];
16                    array[j + 1] = temp;
17                }
18
19
20     }
21
22     public static void main(String args[]) {
23         int[] data = { -2, 45, 0, 11, -9 };
24         Main.bubbleSort(data);
25
26         System.out.println("Sorted Array in Ascending Order:");
27         System.out.println(Arrays.toString(data));
28     }
29 }
```

The 'DEBUG CONSOLE' at the bottom shows the output: 'Sorted Array in Ascending Order: [-9, -2, 0, 11, 45]'. The status bar at the bottom indicates 'Ln 15, Col 35', 'Spaces: 4', 'UTF-8', 'CRLF', 'Java', and 'Prettier'.

8. Kth largest element



The screenshot shows the same IDE with the 'Kthlargest.java' file open. The code is as follows:

```
1 import java.util.Arrays;
2
3 class Klargest {
4
5     public static int findKthLargest(Integer[] arr, int k) {
6         Arrays.sort(arr); // Sort the array in ascending order
7         int n = arr.length;
8         return arr[n - k]; // Access the kth largest element
9     }
10
11     public static void main(String[] args) {
12         int K = 3; // Find the 3rd largest element
13         Integer[] arr = new Integer[]{24, 35, 46, 58};
14         System.out.println("The " + K + "rd largest element is: " + findKthLargest(arr, K));
15     }
16 }
17 }
```

The 'DEBUG CONSOLE' at the bottom shows the output: 'The 3rd largest element is: 35'. The status bar at the bottom indicates 'Ln 15, Col 35', 'Spaces: 4', 'UTF-8', 'CRLF', 'Java', and 'Prettier'.

9.Prime number or not

The screenshot shows a Windows 11 desktop with a taskbar at the bottom. The taskbar includes the Start button, a search bar, and several pinned application icons: File Explorer, Edge, VS Code, Word, PowerPoint, and a custom icon. The system tray on the right shows the date and time as 12:28 PM on 11/10/2024, along with icons for network, volume, and battery.

The main application is Visual Studio Code, which is open with a Java project named 'javaprogs'. The Explorer sidebar on the left shows the project structure with 'Main.java' selected. The editor displays the following Java code:

```

1
2 class Main {
3
4     public static void main(String[] args) {
5
6         int num = 29;
7         boolean flag = false;
8         if (num == 0 || num == 1) {
9             flag = true;
10        }
11
12        for (int i = 2; i <= num / 2; ++i) {
13
14
15            if (num % i == 0) {
16                flag = true;
17                break;
18            }
19        }
20
21        if (!flag)
22            System.out.println(num + " is a prime number.");
23        else
24            System.out.println(num + " is not a prime number.");
25    }
26}

```

Below the editor is the TERMINAL panel, which shows the output of the compilation and execution commands:

```

Prime.java:1: error: class Main is public, should be declared in a file named Main.java
public class Main {
      ^
1 error
PS C:\Users\prasa\OneDrive\Documents\javaprogs> javac Main.java
PS C:\Users\prasa\OneDrive\Documents\javaprogs> java Main
29 is a prime number.
PS C:\Users\prasa\OneDrive\Documents\javaprogs>

```

The status bar at the bottom of VS Code indicates the current cursor position as 'Ln 2, Col 3', the encoding as 'UTF-8', and the language as 'Java'. The bottom of the screen shows the Windows taskbar with the temperature at 31°C and the weather as 'Partly sunny'.

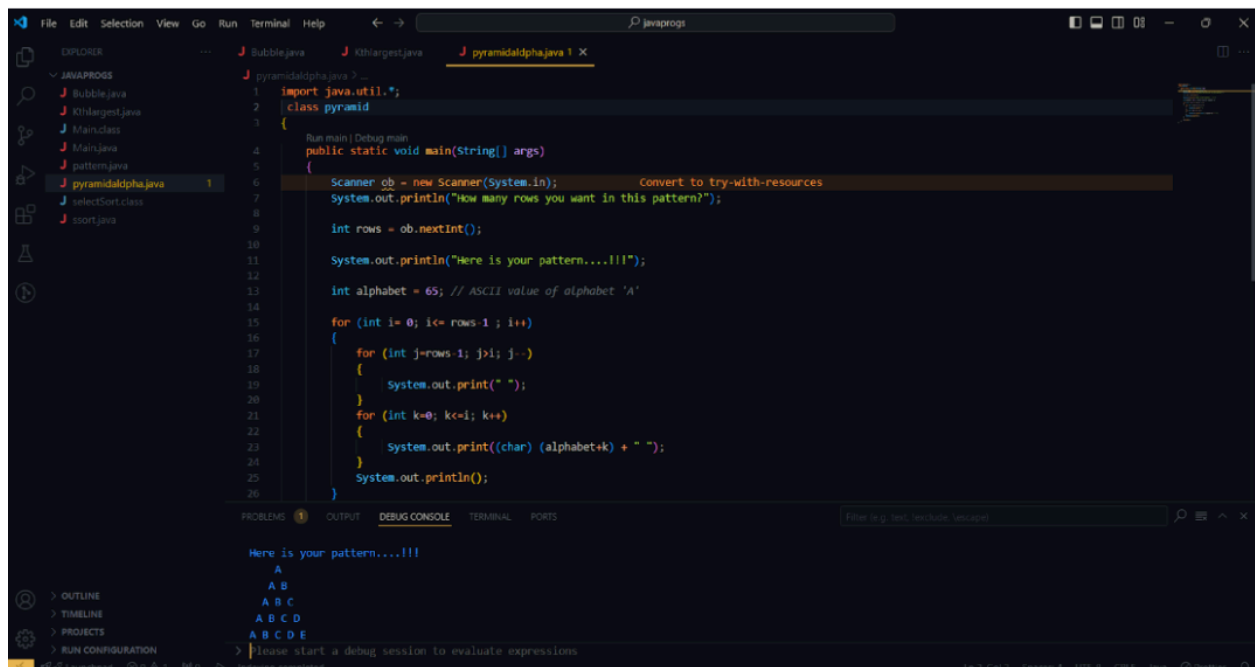
10. Pattern program

The screenshot shows an IDE with a dark theme. The Explorer panel on the left lists files: Bubble.java, Kthlargest.java, Main.class, Main.java, pattern.java (selected), selectSort.class, and ssort.java. The main editor displays the following code in pattern.java:

```

1  class HelloWorld {
2      Run main | Debug main
3      public static void main(String[] args) {
4          for(int i=0; i<5; i++){
5              System.out.println("");//
6          }
7          System.out.println(" ");
8      }
9  }
10 }
```

11.pyramid alphabits



The screenshot shows an IDE with a Java file named `pyramidalphajava.java`. The code defines a `pyramid` class with a `main` method that prompts the user for the number of rows and prints a pyramid pattern of alphabets. The output window shows the pattern for 5 rows.

```
1 import java.util.*;
2 class pyramid
3 {
4     public static void main(String[] args)
5     {
6         Scanner ob = new Scanner(System.in);
7         System.out.println("How many rows you want in this pattern?");
8
9         int rows = ob.nextInt();
10
11        System.out.println("Here is your pattern....!!!");
12
13        int alphabet = 65; // ASCII value of alphabet 'A'
14
15        for (int i=0; i<= rows-1; i++)
16        {
17            for (int j=rows-1; j>1; j--)
18            {
19                System.out.print(" ");
20            }
21            for (int k=0; k<=i; k++)
22            {
23                System.out.print((char) (alphabet+k) + " ");
24            }
25            System.out.println();
26        }
27    }
28 }
```

Here is your pattern....!!!

```

  A
 A B
A B C
A B C D
A B C D E
```