

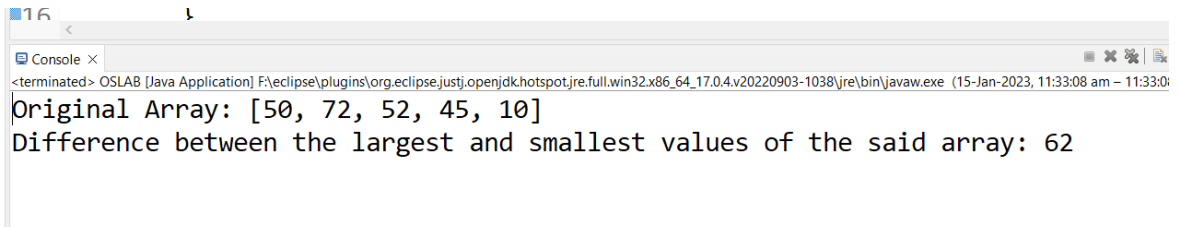
OS LAB EXAM

Tejas Khadke 220960920038

1)

```
package com.os.main;
import java.util.Arrays;
public class OSLAB {
    public static void main(String[] args)
    {
        int[] array_nums = {50, 72, 52, 45,
10};
        System.out.println("Original Array:
"+Arrays.toString(array_nums));
        int max_val = array_nums[0];
        int min = array_nums[0];
        for(int i = 1; i < array_nums.length;
i++)
        {
            if(array_nums[i] > max_val)
                max_val = array_nums[i];
            else if(array_nums[i] < min)
                min = array_nums[i];
        }
        System.out.println("Difference between
the largest and smallest "
+ "values of the said array:
"+(max_val-min));
    }
```

}



The screenshot shows a Java console window titled "Console x". The window contains the following text:
<terminated> OSLAB [Java Application] F:\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.4.v20220903-1038\jre\bin\javaw.exe (15-Jan-2023, 11:33:08 am - 11:33:08 am)
Original Array: [50, 72, 52, 45, 10]
Difference between the largest and smallest values of the said array: 62

2)

day5 > C child.c > main()

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <sys/types.h>
4  #include <unistd.h>
5
6  // Driver code
7  int main()
8  {
9      int arr[] = {1, 2, 3, 4, 5};
10     pid_t pid;
11
12     // fork a child process
13     pid = fork();
14
15     if (pid < 0)
16     {
17         fprintf(stderr, "Fork Failed");
18         return 1;
19     }
20     else if (pid == 0)
21     {
22         // child process
23         int n = sizeof(arr) / sizeof(arr[0]);
24         for (int i = 0; i < n; i++)
25             printf("%d ", arr[i]);
26
27         printf("Child Process is Finished \n");
28     }
29     else
30     {
31         // parent process
32         wait(NULL);
33         printf("Parent Process is Finished\n");
34     }
35
36     return 0;
37 }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Code-
Thread Function: Thread 1
osboxes@osboxes:~/one/day5$ cd "/home/osboxes/one/day5/" && gcc child.c -o child && "/home/osboxes/one/day5/"child
child.c: In function 'main':
child.c:32:9: warning: implicit declaration of function 'wait' [-Wimplicit-function-declaration]
   32 |         wait(NULL);
      |         ^~~~
1 2 3 4 5 Child Process is Finished
Parent Process is Finished
osboxes@osboxes:~/one/day5$
```

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>

// Driver code
int main()
{
    int arr[] = {1, 2, 3, 4, 5};
    pid_t pid;

    // fork a child process
    pid = fork();

    if (pid < 0)
    {
        fprintf(stderr, "Fork Failed");
        return 1;
    }
    else if (pid == 0)
    {
        // child process
        int n = sizeof(arr) / sizeof(arr[0]);
        for (int i = 0; i < n; i++)
            printf("%d ", arr[i]);

        printf("Child Process is Finished \n");
    }
    else
    {
        // parent process
        wait(NULL);
        printf("Parent Process is Finished\n");
    }

    return 0;
}
```