

DAILY ONLINE ACTIVITIES SUMMARY

| | | | |
|--|---|-----------------|-------------|
| Date: | 22-06-2020 | Name: | Deepika K V |
| Sem & Sec | 8 th sem 'A' sec | USN: | 4AL16CS030 |
| Online Test Summary | | | |
| Subject | SMS | | |
| Max. Marks | 60 | Score | - |
| Certification Course Summary | | | |
| Course | Learn Projects in Javascript and JQuery | | |
| Certificate Provider | eduonix | Duration | 9 hrs |
| Coding Challenges | | | |
| Problem Statement: Write a python program to check the given binary tree is a valid binary search tree(BST) or not. | | | |
| Status: SUBMITTED | | | |
| Uploaded the report in Github | | YES | |
| If yes Repository name | | Codes | |
| Uploaded the report in slack | | YES | |

Online test details:

Certification Course Details:

The screenshot shows the Eduonix website interface. The header includes the Eduonix logo, a search bar with the text "What you want to learn today?", and navigation links for "LIFETIME MEMBERSHIP" and "OFFER ZONE". The user is logged in as "Hi Deepika". The main content area features a large banner for "Projects Using JavaScript & jQuery" with the subtitle "Project 1 Intro - Simple JavaScript Quiz". Below the banner, the text "Project 1 Intro" is displayed, followed by "From the course: Learn Projects In JavaScript And JQuery" and a "Generate Certificate" button. On the right side, there is a sidebar with tabs for "Contents", "Q&A", "Notes", and "Review". The "Contents" tab is active, showing a list of lectures under the heading "All Lectures (50)". The list includes:

- 1: Introduction (1/1 Lectures Completed)
- 2: Simple JavaScript Quiz (5/5 Lectures Completed)

Under the "2: Simple JavaScript Quiz" section, the following lectures are listed:

- 1 Course Intro
- 2 Project 1 Intro (highlighted)
- 3 JavaScript Fundamentals
- 4 Quiz File Structure and HTML
- 5 Quiz CSS
- 6 Quiz JavaScript Logic

Coding Challenge:

```
#include<stdio.h>
```

```
int check_palindrome(int n)
{
    int div = 1;
    while (n / div >= 10)
        div *= 10;

    while (n != 0)
    {
        int first = n / div;
        int last = n % 10;

        // If first and last digits are not same then return false
    }
}
```

```

        if (first != last)
            return -1;

        // Removing the leading and trailing digits from the number
        n = (n % div) / 10;

        // Reducing divisor by a factor of 2 as 2 digits are dropped
        div = div / 100;
    }
    return 1;
}

int large_palindrome(int A[], int n)
{
    int i;
    // Sort the array
    for(int i=0; i<n; i++)
    {
        for(int j=i; j<= n; j++)
        {
            if(A[i] >A [j])
            {
                int temp = A[i];
                A[i] = A[j];
                A[j] = temp;
            }
        }
    }

    for(int i=0; i<n; i++)
    {
        printf("%d ", A[i]);
    }

    for (i=n-1; i >= 0; i--)
    {
        if (check_palindrome(A[i]) == 1)
            return A[i];
    }
    return -1;
}

```

```
}  
int main()  
{  
    int a[15], n, i;  
    printf("Enter the number of entries: \n");  
    scanf("%d", &n);  
    printf("Enter the elements: \n");  
    for(i=0; i<n; i++)  
        scanf("%d", &a[i]);  
    printf("\n Largest Palindrome: %d", large_palindrome(a, n));  
    return 0;  
}
```