

**DAILY ONLINE ACTIVITIES SUMMARY**

<b>Date:</b>	<b>25/06/2020</b>	<b>Name:</b>	<b>Deepa</b>
<b>Sem &amp; Sec</b>	<b>8<sup>th</sup> Sem</b>	<b>USN:</b>	<b>4AL16CS029</b>
<b>Online Test Summary</b>			
<b>Subject</b>	<b>System Model-ling and Simulation</b>		
<b>Max. Marks</b>	<b>30</b>	<b>Score</b>	<b>- -</b>
<b>Certification Course Summary</b>			
<b>Course</b>	<b>The complete wi-fi hacking course: beginner to advance 2019</b>		
<b>Certificate Provider</b>	<b>udemy.com/</b>	<b>Duration</b>	<b>3 hrs</b>
<b>Coding Challenges</b>			
<b>Problem Statement: 1) Write a C program to find if an array is a subset of another array</b>			
<b>Status: Completed</b>			
<b>Uploaded the report in Github</b>		<b>Yes</b>	
<b>If yes Repository name</b>		<b>Daily_report</b>	
<b>Uploaded the report in slack</b>		<b>yes</b>	

## Online Test Details:

--

## Certification Course Details:

- ✓ 1. About course updates and supports you will get

2min

### Section 2: Ethics of Hacking and Wi-Fi Protocols

2 / 2 | 9min

- ✓ 2. Ethics of Hacking

3min

 Resources ▾

- ✓ 3. Different protocols of Wi-Fi

6min

### Section 3: Getting started with Wifislax

2 / 2 | 9min

- ✓ 4. Getting ready for the installation

4min

- ✓ 5. Installing Wifislax

5min

### Section 4: Breaking WEP

3 / 3 | 19min

- ✓ 6. Enabling WEP Encryption

3min

- ✓ 7. Breaking WEP encryption with Active Client

6min

- ✓ 8. Breaking WEP encryption with Fake client

✓ 9. Configuring WPA/WPA2 and enabling WPS

▶ 2min

✓ 10. Methodology of cracking WPS

▶ 3min

✓ 11. Cracking WPS enabled WPA/WPA2 encryption

▶ 7min

**Section 6: Breaking WPA/WPA2 (WPS Disabled)**



4 / 4 | 19min

✓ 12. Deauthenticating client(s) and collecting WPA handshakes

▶ 3min

✓ 13. Creating effective password dictionary

▶ 6min

✓ 14. Cracking WPA/WPA2 encryption

▶ 4min

✓ 15. Cracking WPA/WPA2 encryption using Rainbow tables.

▶ 6min

**Section 7: Breaking WEP on Windows**



1 / 2 | 9min

✓ 16. Gathering necessary hacking tools

▶ 3min



## Coding Challenges Details:

### Program 1:

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

```
int isSubset(int arr1[], int arr2[],
```

```
    int m, int n)
```

```
{
```

```
    int i = 0;
```

```
    int j = 0;
```

```
    for (i = 0; i < n; i++)
```

```
    {
```

```
        for (j = 0; j < m; j++)
```

```
        {
```

```
            if(arr2[i] == arr1[j])
```

```
                break;
```

```
        }
```

```
    if (j == m)
```

```
        return 0;
```

```
}
```

```
return 1;
```

```
}
```

```
int main()
```

```
{
```

```
    int m,n;
```

```
    printf("\nEnter the size of array 1 : ");
```

```
    scanf("%d", &m);
```

```
    printf("\nEnter the size of array 2 : ");
```

```
    scanf("%d", &n);
```

```
    int arr1[m],arr2[n];
```

```
    int i;
```

```
    printf("\nEnter the array 1 elements : ");
```

```
    for(i=0;i<m;i++)
```

```
    {
```

```
        scanf("%d",&arr1[i]);
```

```
    }
```

```
    printf("\nEnter the array 2 elements : ");
```

```
    for(i=0;i<n;i++)
```

```
    {
```

```
        scanf("%d",&arr2[i]);
```

```
    }
```

```
    if(isSubset(arr1, arr2, m, n))
```

```
printf("\nArray2 is a subset of Array1\n ");
```

```
else
```

```
printf("\nArray2 is not a subset of Array1\n");
```

```
getchar();
```

```
return 0;
```

```
}
```