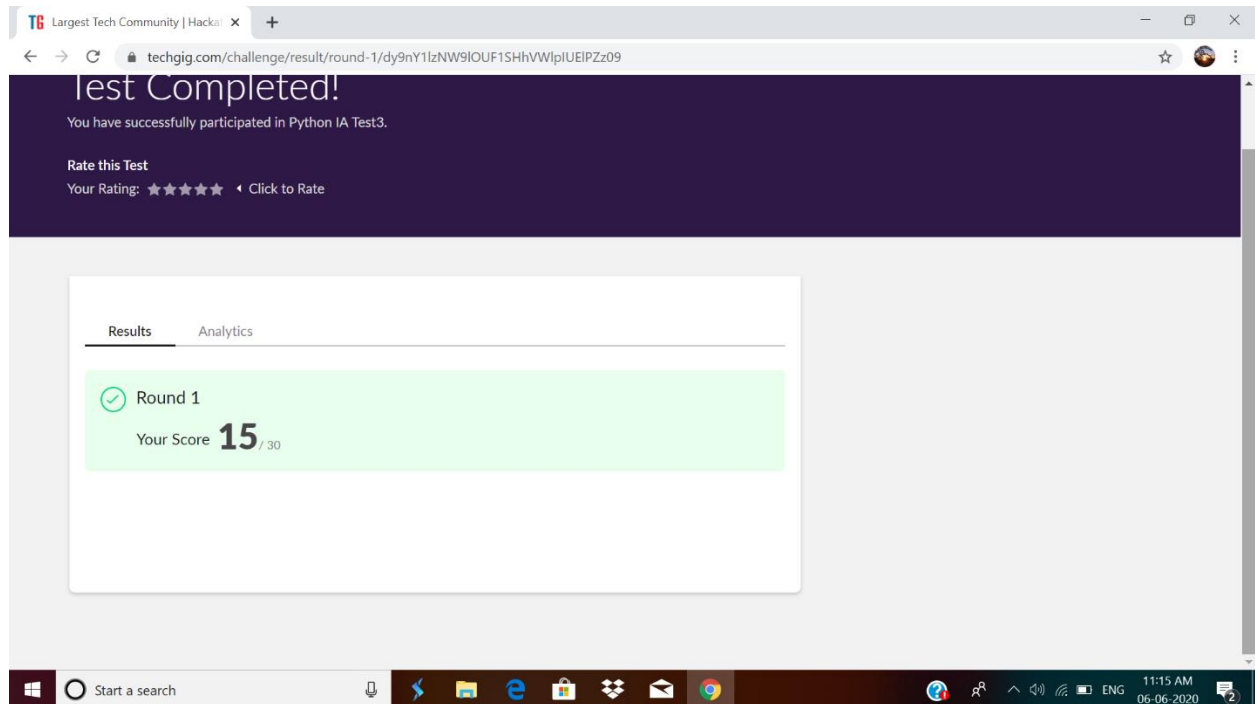


DAILY ONLINE ACTIVITIES SUMMARY

| | | | |
|--|------------------------------|---|-----------------|
| Date: | 06-06-2020 | Name: | Anvitha Poojary |
| Sem & Sec | 6A | USN: | 4AL17CS008 |
| Online Test Summary | | | |
| Subject | PAP | | |
| Max. Marks | 30 | Score | 15 |
| Certification Course Summary | | | |
| Course | Machine Learning with Python | | |
| Certificate Provider | COGNITIVE CLASS .ai | Duration | 12hr |
| Coding Challenges | | | |
| Problem Statement: 1. Write a program in C to rotate an array by N positions. 2. Write a Python program to perform Cyclic Redundancy Check 3. Write a Python program to count the number of strings, provided string length is 2 or more and the first and last character are same from a given list of strings. | | | |
| Status: completed | | | |
| Uploaded the report in Github | | Yes | |
| If yes Repository name | | https://github.com/anvithapo99/Daily-Report | |
| Uploaded the report in slack | | Yes | |

Online test details:

Subject: PAP



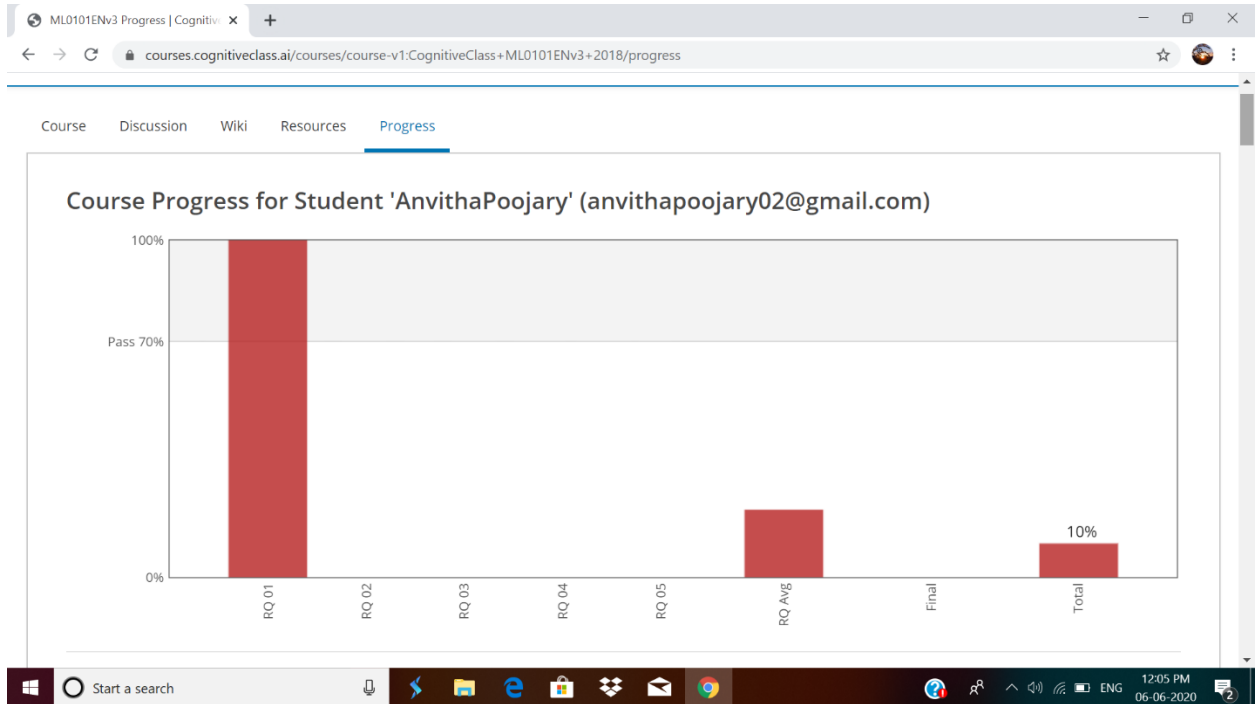
Certification course details:

Machine Learning with Python

Today I have learnt following topics:

- Python libraries for machine
- Scipy
- Matplotlib
- Pandas
- Scikit learn

- Supervised vs unsupervised
- What is clustering



Coding Challenges Details:

1. Write a program in C to rotate an array by N positions.

Expected Output :

The given array is : 0 3 6 9 12 14 18 20 22 25 27

Enter the Position N from where you want to rotate: 4

From 4th position the values of the array are : 12 14 18 20 22 25 27

Before 4th position the values of the array are : 0 3 6 9

After rotating from 4th position the array is:

12 14 18 20 22 25 27 0 3 6 9

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

```
void shiftArr1Pos(int *arr1, int arrSize)
```

```
{
```

```

int i, temp;
temp = arr1[0];
for(i = 0; i < arrSize-1; i++)
{
    arr1[i] = arr1[i+1];
}
arr1[i] = temp;
}
void arr1Rotate(int *arr1, int arrSize, int rotFrom)
{
    int i;
    for(i = 0; i < rotFrom; i++)
    {
        shiftArr1Pos(arr1, arrSize);
    }
    return;
}
int main()
{
    int arr1[] = {0,3,6,9,12,14,18,20,22,25,27};
    int ctr = sizeof(arr1)/sizeof(arr1[0]);
    int i;
    printf("The given array is : ");
    for(i = 0; i < ctr; i++)
    {
        printf("%d ", arr1[i]);
    }
    printf("\n");
    printf("From 4th position the values of the array are : ");
    for(i = 4; i < ctr; i++)
    {
        printf("%d ", arr1[i]);
    }
    printf("\n");
    printf("Before 4th position the values of the array are : ");
    for(i = 0; i < 4; i++)
    {
        printf("%d ", arr1[i]);
    }
    printf("\n");
    arr1Rotate(arr1, ctr, 4);
    printf("\nAfter rotating from 4th position the array is: \n");

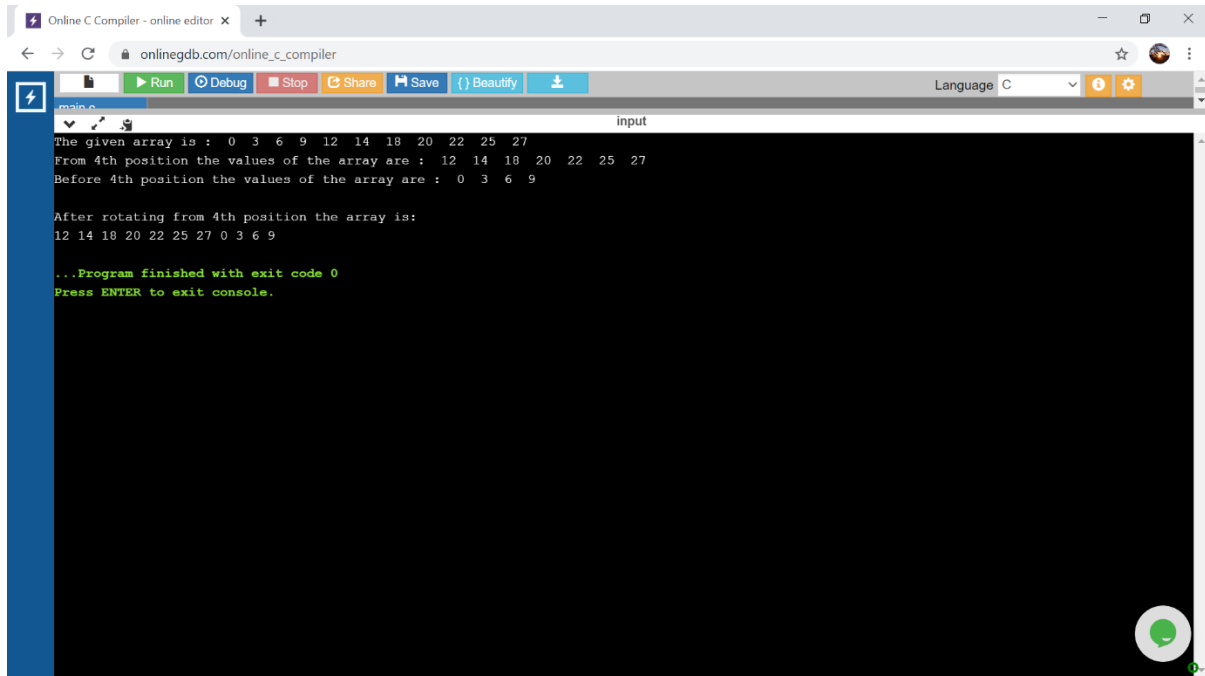
```

```

    for(i = 0; i<ctr; i++)
    {
        printf("%d ", arr1[i]);
    }
    return 0;
}

```

Output:



```

input
The given array is : 0 3 6 9 12 14 18 20 22 25 27
From 4th position the values of the array are : 12 14 18 20 22 25 27
Before 4th position the values of the array are : 0 3 6 9

After rotating from 4th position the array is:
12 14 18 20 22 25 27 0 3 6 9

...Program finished with exit code 0
Press ENTER to exit console.

```

2. Write a Python program to perform Cyclic Redundancy Check

CRC uses Generator Polynomial which is available on both sender and receiver side. An example generator polynomial is of the form like $x^3 + x + 1$. This generator polynomial represents key 1011. Another example is $x^2 + 1$ that represents key 101.

Data word to be sent - 100100

Key - 1101 [Or generator polynomial $x^3 + x^2 + 1$]

```
def xor(a, b):
```

```
    # initialize result
```

```
result = []
```

```
# Traverse all bits, if bits are
```

```
# same, then XOR is 0, else 1
```

```
for i in range(1, len(b)):
```

```
    if a[i] == b[i]:
```

```
        result.append('0')
```

```
    else:
```

```
        result.append('1')
```

```
return ''.join(result)
```

```
def mod2div(divident, divisor):
```

```
    pick = len(divisor)
```

```
    tmp = divident[0 : pick]
```

```
    while pick < len(divident):
```

```
        if tmp[0] == '1':
```

```
tmp = xor(divisor, tmp) + dividend[pick]
```

```
else:
```

```
tmp = xor('0'*pick, tmp) + dividend[pick]
```

```
pick += 1
```

```
if tmp[0] == '1':
```

```
tmp = xor(divisor, tmp)
```

```
else:
```

```
tmp = xor('0'*pick, tmp)
```

```
checkword = tmp
```

```
return checkword
```

```
def encodeData(data, key):
```

```
l_key = len(key)
```

```
appended_data = data + '0'*(l_key-1)
```

```
remainder = mod2div(appended_data, key)
```

```
codeword = data + remainder
```

```
print("Remainder : ", remainder)
```

```
print("Encoded Data (Data + Remainder) : ", codeword)
```

```
data = "100100"
```

```
key = "1101"
```

```
encodeData(data, key)
```

output:



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\hp\Desktop\program\cylinder.py =====
Remainder : 001
Encoded Data (Data + Remainder) : 100100001
>>>
```

3. Description:

Write a Python program to count the number of strings, provided string length is 2 or more and the first and last character are same from a given list of strings.

Eg:

Input

```
list1['hia', 'aba' , '363']
```

Output:

Number of strings with first and last cahracter is same: 2

```
def match_words(words):
```

```
    ctr = 0
```

```
    for word in words:
```

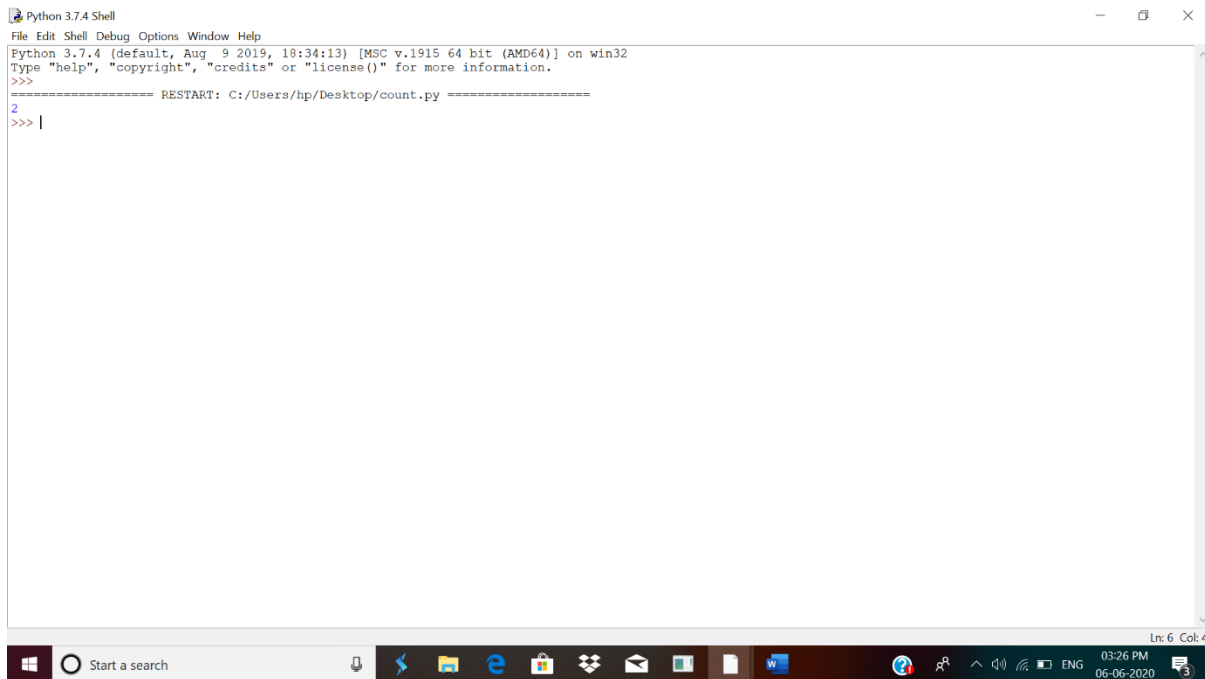
```
        if len(word) > 1 and word[0] == word[-1]:
```

```
            ctr += 1
```

```
    return ctr
```

```
print(match_words(['hia', 'aba' , '363']))
```

output:



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/hp/Desktop/count.py =====
2
>>> |
```