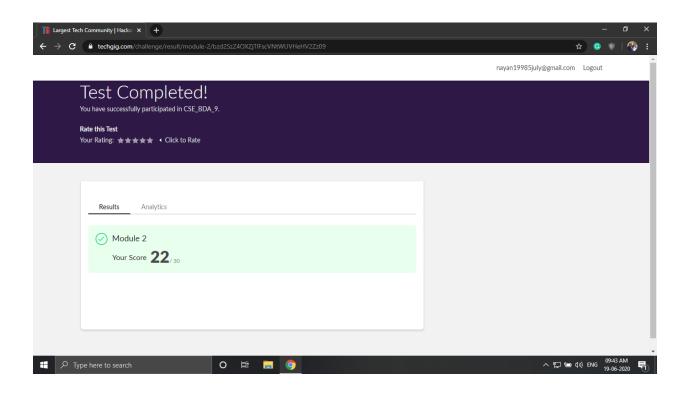
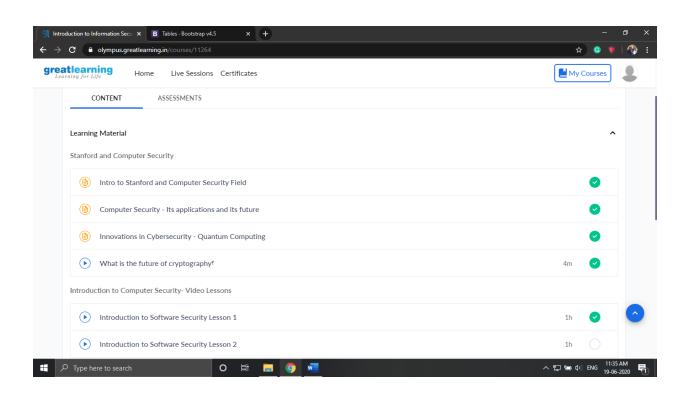
## **DAILY ONLINE ACTIVITIES SUMMARY**

Date:	19-06-2020		Name:	Nayan. P. Joshi		
Sem & Sec	8 <sup>th</sup> Sem A		USN:	4AL16CS058		
Online Test Summary						
Subject	ect Big Data Analytics					
Max. Marks	30		Score 22			
Certification Course Summary						
Course	Course Introduction to Information Security					
Certificate Provider		Great learning academy	Duration		6hrs	
Coding Challenges						
<b>Problem Statement:</b> Write a C Program to rotate a Matrix by 90 Degree in Clockwise or Anticlockwise Direction.						
Status: Solved						
Uploaded the report in GitHub			yes			
If yes Repository name			nayan1998			
Uploaded th	e report i	n slack	yes			





Write a C Program to rotate a Matrix by 90 Degree in Clockwise or Anticlockwise Direction

```
#include <bits/stdc++.h>
#define N 3
using namespace std;
// Function to reverse rows of the matrix
void reverse_rows(int arr[N][N])
  int k;
  for (int i = 0; i < N; i++)
  {
     k = N-1;
     for (int j = 0; j < k; j++)
        swap(arr[i][j], arr[i][k]);
        k--;
     }}}
// Transpose of a matrix
void transpose1(int arr[N][N])
{
  for (int i = 0; i < N; i++)
     for (int j = i; j < N; j++)
        swap(arr[i][j], arr[j][i]);
}
// print the matrix
void print_matrix1(int mat[N][N])
  for (int i = 0; i < N; i++)
```

```
for (int j = 0; j < N; j++)
         printf("%d\t", mat[i][j]);
     printf("\n");
  printf("\n");
void reverse_column(int arr[N][N])
   int k;
  for (int i = 0; i < N; i++)
     k = N-1;
     for (int j = 0; j < k; j++)
         swap(arr[j][i], arr[k][i]);
        k--;
     }}}
// Transpose of a matrix
void transpose2(int arr[N][N])
  for (int i = 0; i < N; i++)
     for (int j = i; j < N; j++)
        swap(arr[i][j], arr[j][i]);
void print_matrix2(int mat[N][N])
  for (int i = 0; i < N; i++)
     for (int j = 0; j < N; j++)
         printf("%d\t", mat[i][j]);
     printf("\n");
```

```
printf("\n");
// Main function
int main()
{
  int mat[N][N] = \{1, 2, 3, 4, 5, 6, 7, 8, 9\};
  printf("The matrix before rotation\n");
  print matrix1(mat);
  transpose1(mat);
  printf("The matrix after rotation - Clockwise\n");
  reverse_rows(mat);
  print_matrix1(mat);
  printf("-----
-\n");
  printf("The matrix before rotation\n");
  print_matrix2(mat);
  transpose2(mat);
  printf("The matrix after rotation - anticlockwise\n");
  reverse column(mat);
  print_matrix2(mat);
  return 0;
```