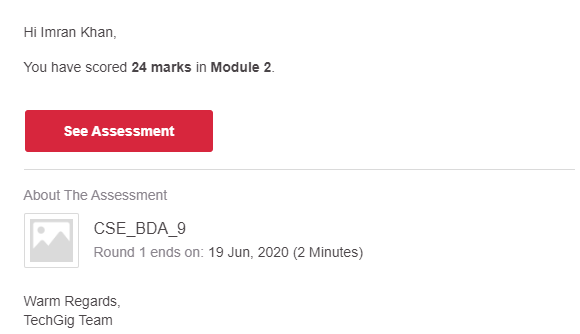
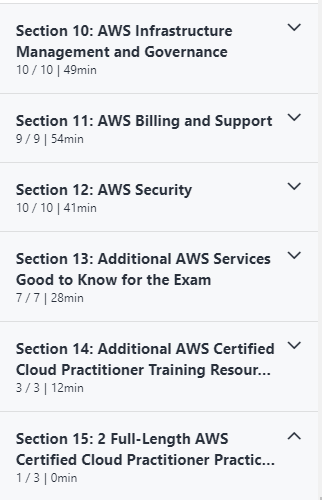
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **19/06/2020** | | | | **Name:** | **Imran Khan** | |
| **Sem & Sec** | **8th A** | | | | **USN:** | **4AL16CS040** | |
| **Online Test Summary** | | | | | | | |
| **Subject** | | **BDA** | | | | | |
| **Max. Marks** | | **30** | | **Score** | | **24** | |
| **Certification Course Summary** | | | | | | | |
| **Course** | **Aws** | | | | | | |
| **Certificate Provider** | | | **udemy** | **Duration** | | | **14 HOURS** |
| **Coding Challenges** | | | | | | | |
| Problem Statement:  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** | | | | **Imran040** | | | |
| **Uploaded the report in slack** | | | | **yes** | | | |

Online test details:



**Certification Course Details**:



**Coding Challenges Details**:

**program1:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
|  | | |
|  | |
| #include <stdio.h> | |
|  | |  |
|  | | // Function to reverse rows of the matrix |
|  | | int main() |
|  | | { |
|  | | int mat[10][10], arr[10][10], m, n, i, j, temp, k; |
|  | | printf("Enter the total number of Rows: "); |
|  | | scanf("%d", &m); |
|  | | printf("Enter the total number of Columns: "); |
|  | | scanf("%d", &n); |
|  | |  |
|  | | printf("Enter the elements:\n"); |
|  | | for (i = 0; i < m; i++) |
|  | | for (j = 0; j < n; j++) |
|  | | scanf("%d", &mat[i][j]); |
|  | |  |
|  | | //---print original matrix--- |
|  | | printf("\nThe matrix before rotation\n"); |
|  | | for (i = 0; i < m; i++) |
|  | | { |
|  | | for (j = 0; j < n; j++) |
|  | | printf("%d\t", mat[i][j]); |
|  | | printf("\n"); |
|  | | } |
|  | |  |
|  | | //---transpose of matrix--- |
|  | | for (int i = 0; i < m; i++) |
|  | | for (int j = i; j < n; j++) |
|  | | { |
|  | | temp = mat[i][j]; |
|  | | mat[i][j] = mat[j][i]; |
|  | | mat[j][i] = temp; |
|  | | } |
|  | |  |
|  | | //---copy matrix transpose |
|  | | for (i = 0; i < m; i++) |
|  | | for (j = 0; j < n; j++) |
|  | | arr[i][j] = mat[i][j]; |
|  | |  |
|  | | //---reverse rows for clockwise rotation-- |
|  | | for (int i = 0; i < m; i++) |
|  | | { |
|  | | k = n-1; |
|  | | for (int j = 0; j < k; j++) |
|  | | { |
|  | | temp = mat[i][j]; |
|  | | mat[i][j] = mat[i][k]; |
|  | | mat[i][k] = temp; |
|  | | k--; |
|  | | } |
|  | | } |
|  | |  |
|  | | //--- print matrix after Clockwise rotation--- |
|  | | printf("\nThe matrix after rotation - Clockwise\n"); |
|  | | for (i = 0; i < m; i++) |
|  | | { |
|  | | for (j = 0; j < n; j++) |
|  | | printf("%d\t", mat[i][j]); |
|  | | printf("\n"); |
|  | | } |
|  | |  |
|  | | //---reverse rows for clockwise rotation-- |
|  | | for (int i = 0; i < m; i++) |
|  | | { |
|  | | k = n-1; |
|  | | for (int j = 0; j < k; j++) |
|  | | { |
|  | | temp = arr[j][i]; |
|  | | arr[j][i] = arr[k][i]; |
|  | | arr[k][i] = temp; |
|  | | k--; |
|  | | } |
|  | | } |
|  | |  |
|  | | //--- print matrix after Anticlockwise rotation--- |
|  | | printf("\nThe matrix after rotation - Anticlockwise\n"); |
|  | | for (i = 0; i < m; i++) |
|  | | { |
|  | | for (j = 0; j < n; j++) |
|  | | printf("%d\t", arr[i][j]); |
|  | | printf("\n"); |
|  | | } |
|  | |  |
|  | | return 0; |
|  | | } |
|  | |  |