**DAILY ONLINE ACTIVITIES SUMMARY**

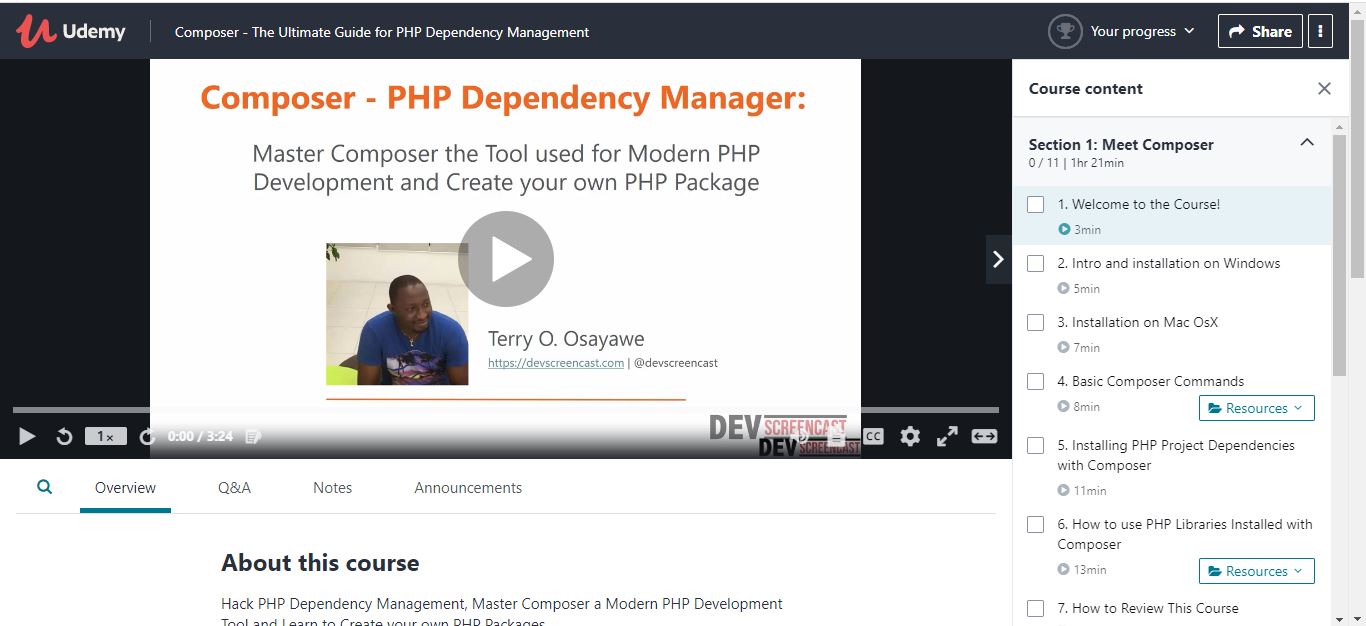
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **28-06-2020** | | | | | **Name:** | **Anix Jugal D’Cunha** | |
| **Sem & Sec** | **8 sem , A sec** | | | | | **USN:** | **4AL16CS013** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **Test was not conducted Today** | | | | | | |
| **Max. Marks** | | **--** | | **Score** | | | **--** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Composer - The Ultimate Guide for PHP Dependency Management**  **(** In – Progress **)** | | | | | | | |
| **Certificate Provider** | | | **Udemy** | | **Duration** | | | 2.5 hours |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:** C program for Binary Search. | | | | | | | | |
| **Status: Competed** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **yes** | | | |
| **If yes Repository name** | | | | | **alvas-education-foundation/dcunhaanixjugal** | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

Online Test Details: (Attach the snapshot and briefly write the report for the same)

Test was not conducted Today

Certification Course Details: (Attach the snapshot and briefly write the report for the same)

( In – Progress )



Coding Challenges Details: (Attach the snapshot and briefly write the report for the same)

## Program-> C program for Binary Search.

|  |
| --- |
| #include <stdio.h> int main() |
|  | { |
|  | int c, first, last, middle, n, search, array[100]; |
|  |  |
|  | printf("Enter number of elements\n"); scanf("%d", &n); |
|  |  |
|  | printf("Enter %d integers\n", n); |
|  |  |
|  | for (c = 0; c < n; c++) scanf("%d", &array[c]); |
|  |  |
|  | printf("Enter value to find\n"); scanf("%d", &search); |
|  |  |
|  | first = 0; last = n - 1; |
|  | middle = (first+last)/2; |
|  |  |
|  | while (first <= last) { |
|  | if (array[middle] < search) first = middle + 1; |
|  | else if (array[middle] == search) { |
|  | printf("%d found at location %d.\n", search, middle+1); break; |
|  | } |
|  | else |
|  | last = middle - 1; |
|  |  |
|  | middle = (first + last)/2; |
|  | } |
|  | if (first > last) |
|  | printf("Not found! %d isn't present in the list.\n", search); |
|  |  |
|  | return 0; |
|  | } |