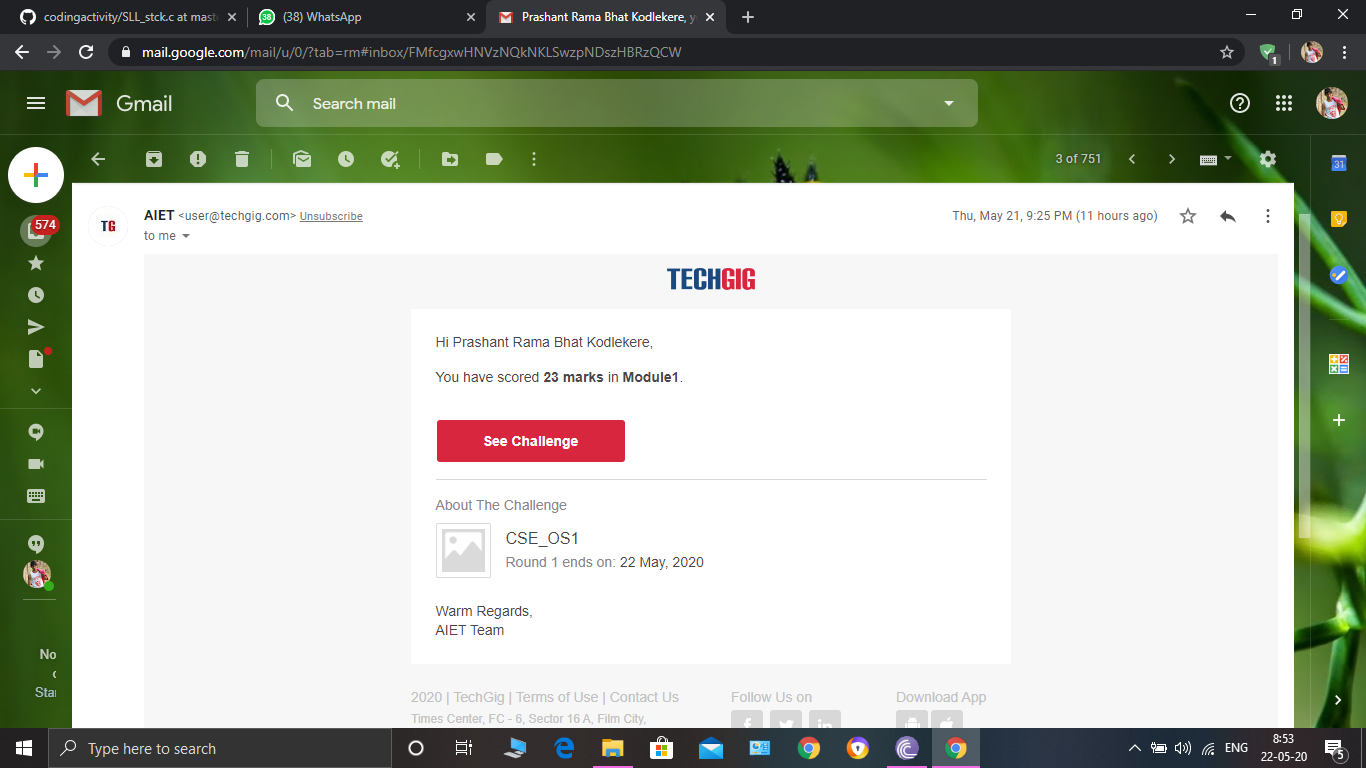
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **22/05/2020** | | | | | **Name:** | **Prashant Rama Bhat Kodlekere** | |
| **Sem & Sec** | **4th sem (B sec)** | | | | | **USN:** | **4AL18CS060** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **OPERATING SYSTEM** | | | | | | |
| **Max. Marks** | | **30** | | **Score** | | | **23** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Data structure and algorithms(3rd session)** | | | | | | | |
| **Certificate Provider** | | | **Udemy** | | **Duration** | | | 28.5 total hours |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:** Write a C Program to implement various operations on Singly Linked List Stack.  Top of Form  Bottom of Form | | | | | | | | |
| **Status:Completed** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **YES** | | | |
| **If yes Repository name** | | | | | **https://github.com/prashantkodlekere/codingactivity** | | | |
| **Uploaded the report in slack** | | | | | **YES** | | | |

Online Test Details:

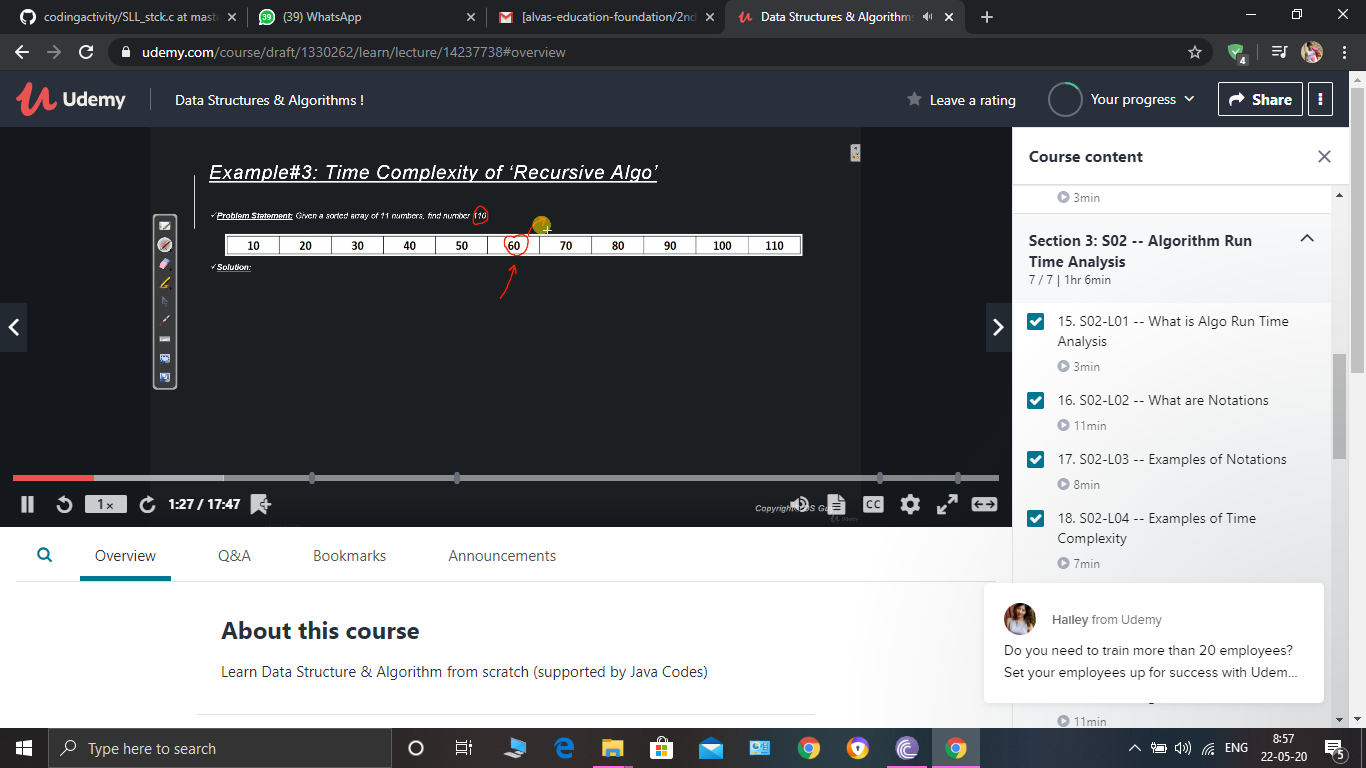


OPERATING SYSTEM Internals was conducted. A total of 30 questionS were Multiple Choice Questions.

The above snapshot is the result sheet which was mailed to us by the Techgig team.

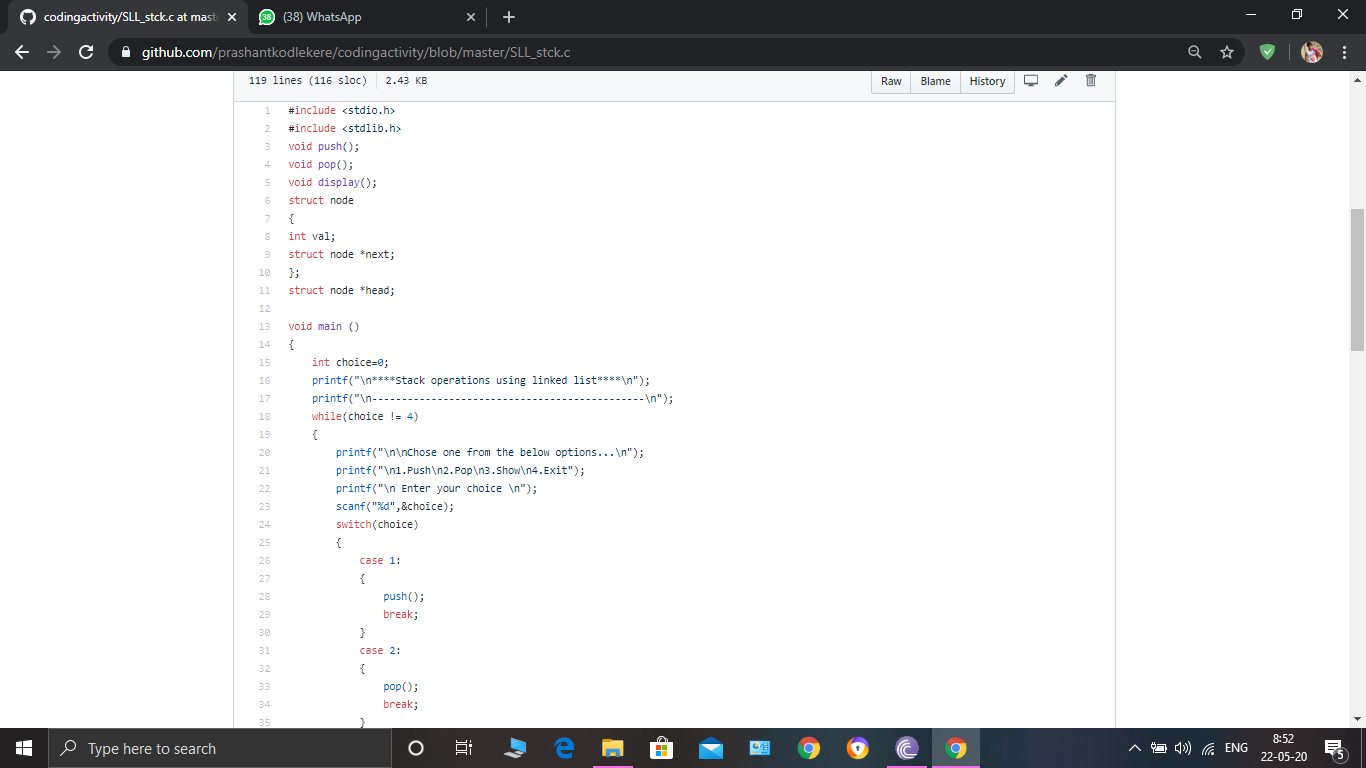
Certification Course Details:

Course name:data structures and algorithms.(3rd session)



In section 3 I learnt what is Run time analysis. Studied Assymtotic notations(Big-oh,Theta,omega)And learnt how to find the time complexity of binary search..and other examples.

Coding Challenges Details:



**The question I took to code is:**

Write a C Program to implement various operations on Singly Linked List Stack.

Top of Form

Bottom of Form

|  |
| --- |
|  |

Hint: First Create a Singly Linked List Stack with the node corresponding to First Element is the base of the stack; and its link field must be always Null.  
When you push First Element, It is the First and it is Base of the stack. Its Link must be Null. top pointer pointing to First. (top = First)  
When you push any element, (No need of checking Stack full case because SLL is dynamic) Create a new node called temp using malloc function and insert the a number into Data field, and Link field must be pointing to top; and move the pointer top to point to temp.  
When you pop, First check for stack Empty. if First == NULL, then Stack Empty. If it is not empty, The pointer temp must be pointing to top. Move the pointer top to top->link. delete temp.  
When you display the stack element, First Check for Stack Empty as in pop operation. If it is not empty, Display all the elements of current stack starting from top to First.

Code: The above snapshot is the code which I have uploaded in my Github repository.