DAILY ONLINE ACTIVITIES SUMMARY

Date:	**22/05/2020**	**Name:**	**Prajwal**
Sem & Sec	**IV sem & B sec**	**USN:**	**4AL18CS057**

Online Test Summary

Subject	**Operating System**		
Max. Marks	**30**	**Score**	**17**

Certification Course Summary

Course	**Machine Learning With Python**		
Certificate Provider	**COGNITIVE CLASS**	**Duration**	**12 hours**

Coding Challenges

Problem Statement: 1. Write a C Program to implement various operations on Singly Linked List Stack.

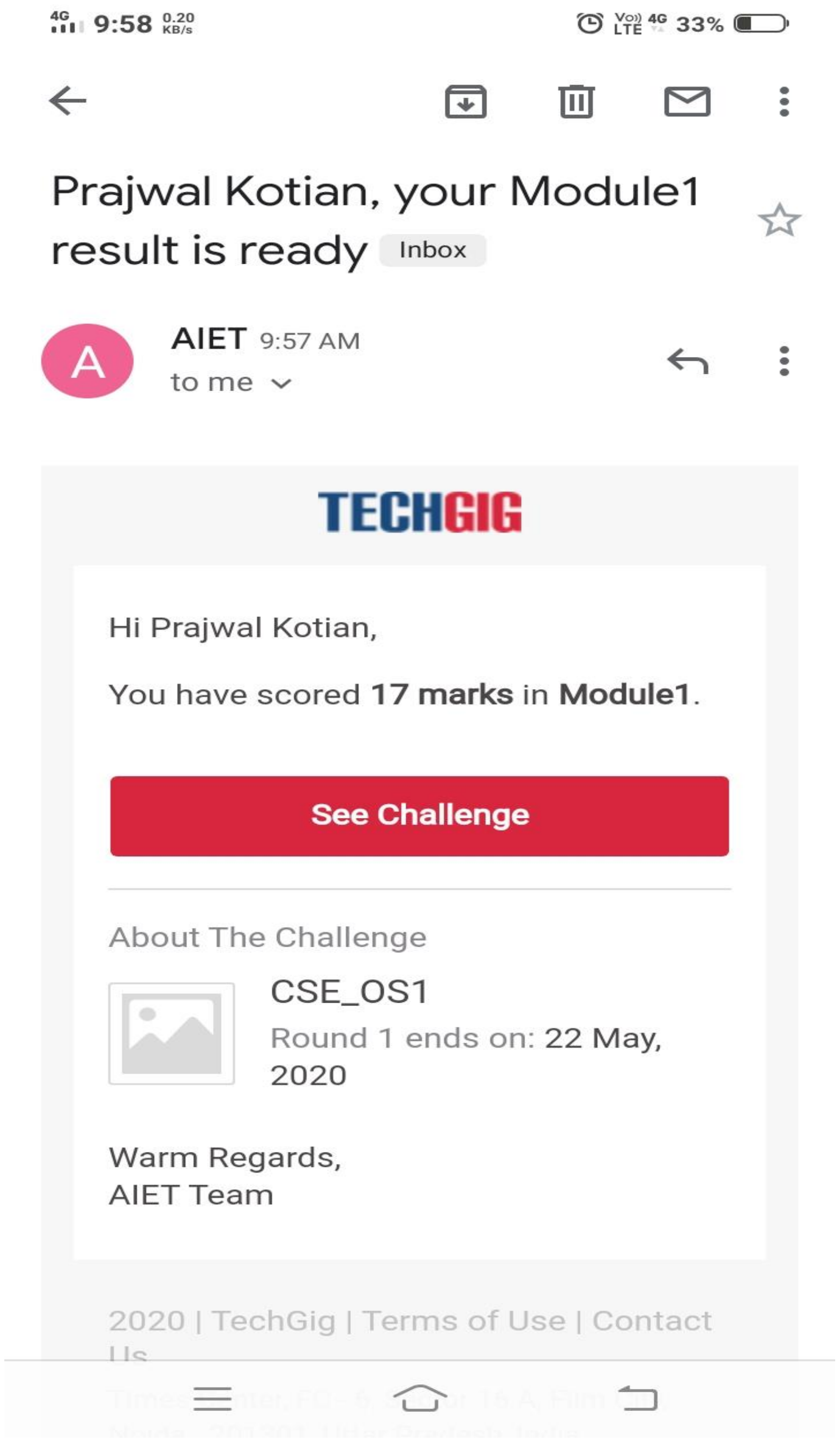
2. Write a C or Java program to implement round robin type of process scheduling

Status: Done

Uploaded the report in Github	**YES**
If yes Repository name	https://github.com/PRAJWALKOTIAN/lockdown-coding
Uploaded the report in slack	**YES**

Online test details

Test was conducted from 09:15 to 09:55 am dated 22 may 2020. The test includes MCQ kind of questions which contains 30 questions of 1 mark each.



9:58 0.20 KB/s

33%

Prajwal Kotian, your Module1 result is ready Inbox

AIET 9:57 AM

to me

TECHGIG

Hi Prajwal Kotian,

You have scored **17 marks** in **Module1**.

See Challenge

About The Challenge

CSE_OS1

Round 1 ends on: 22 May, 2020

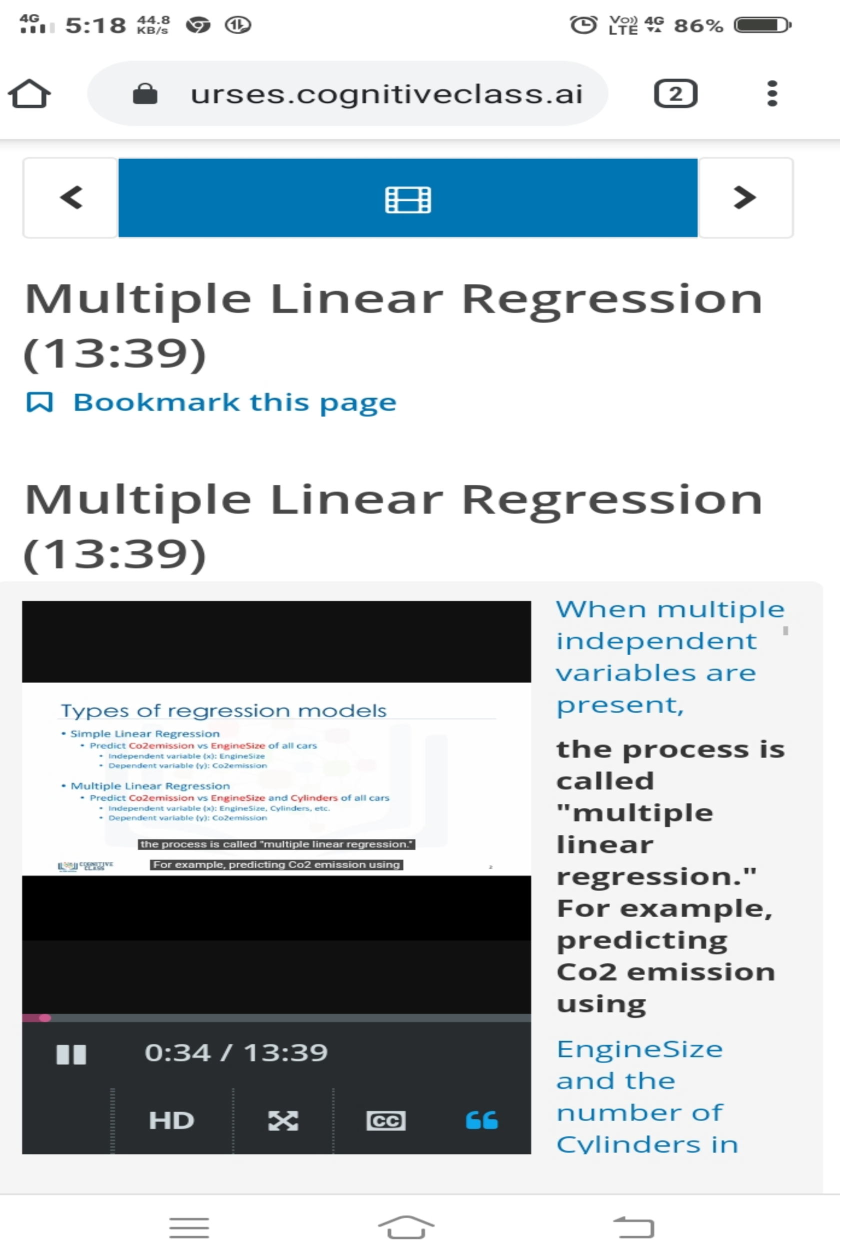
Warm Regards,
AIET Team

2020 | TechGig | Terms of Use | Contact Us

Times Center, FC - 6, Sector 16 A, Film City,
Noida - 201301, Uttar Pradesh, India

Certification Course Details

The cource I have choosen is MACHINE LEARNING WITH PYTHON in this I studied the types of regression models they are multiple linear regression and simple linear regression



Multiple Linear Regression (13:39)

Bookmark this page

Multiple Linear Regression (13:39)

Types of regression models

- Simple Linear Regression
  - Predict Co2emission vs EngineSize of all cars
    - Independent variable (x): EngineSize
    - Dependent variable (y): Co2emission
- Multiple Linear Regression
  - Predict Co2emission vs EngineSize and Cylinders of all cars
    - Independent variable (x): EngineSize, Cylinders, etc.
    - Dependent variable (y): Co2emission

the process is called "multiple linear regression."

For example, predicting Co2 emission using

When multiple independent variables are present,

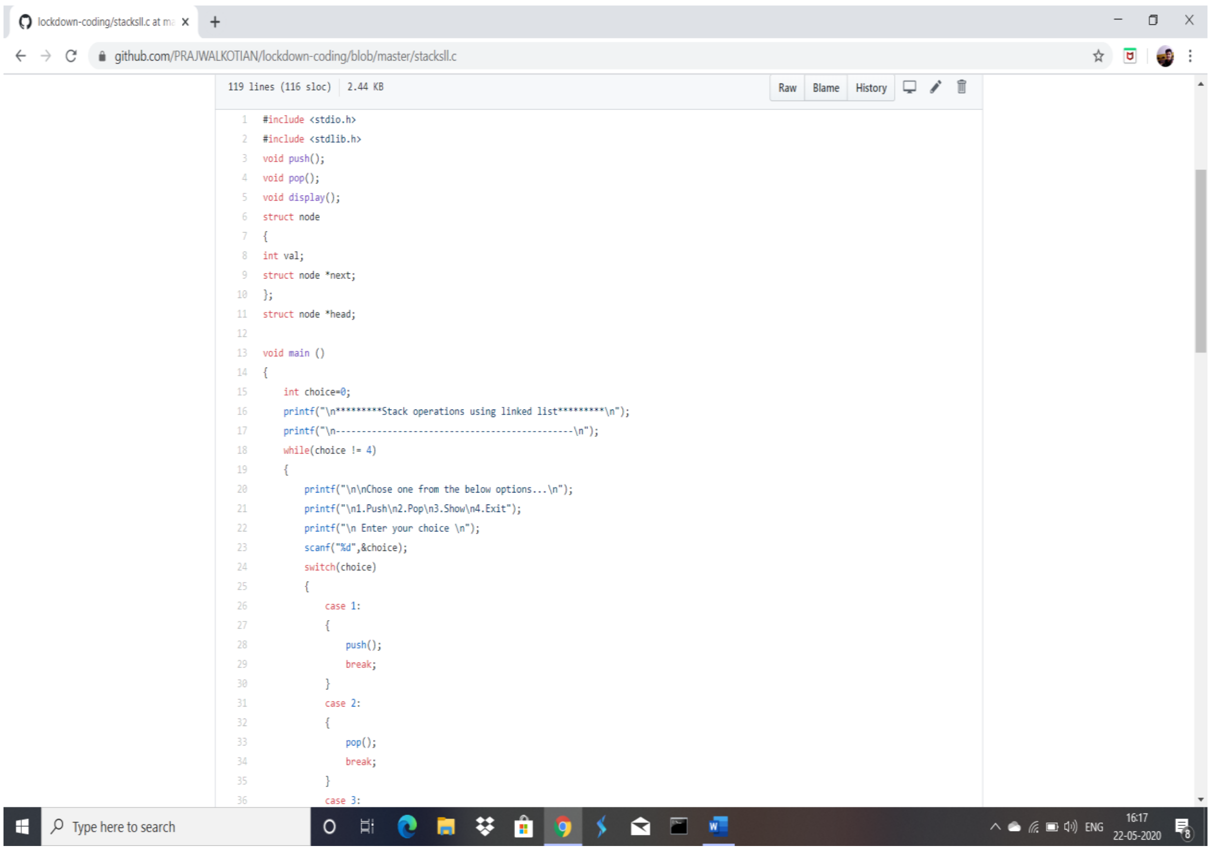
the process is called "multiple linear regression." For example, predicting Co2 emission using

EngineSize and the number of Cylinders in

<u>Coding Challenges Details</u>

The bellow codes are there on my github repository https://github.com/PRAJWALKOTIAN/lockdown-coding

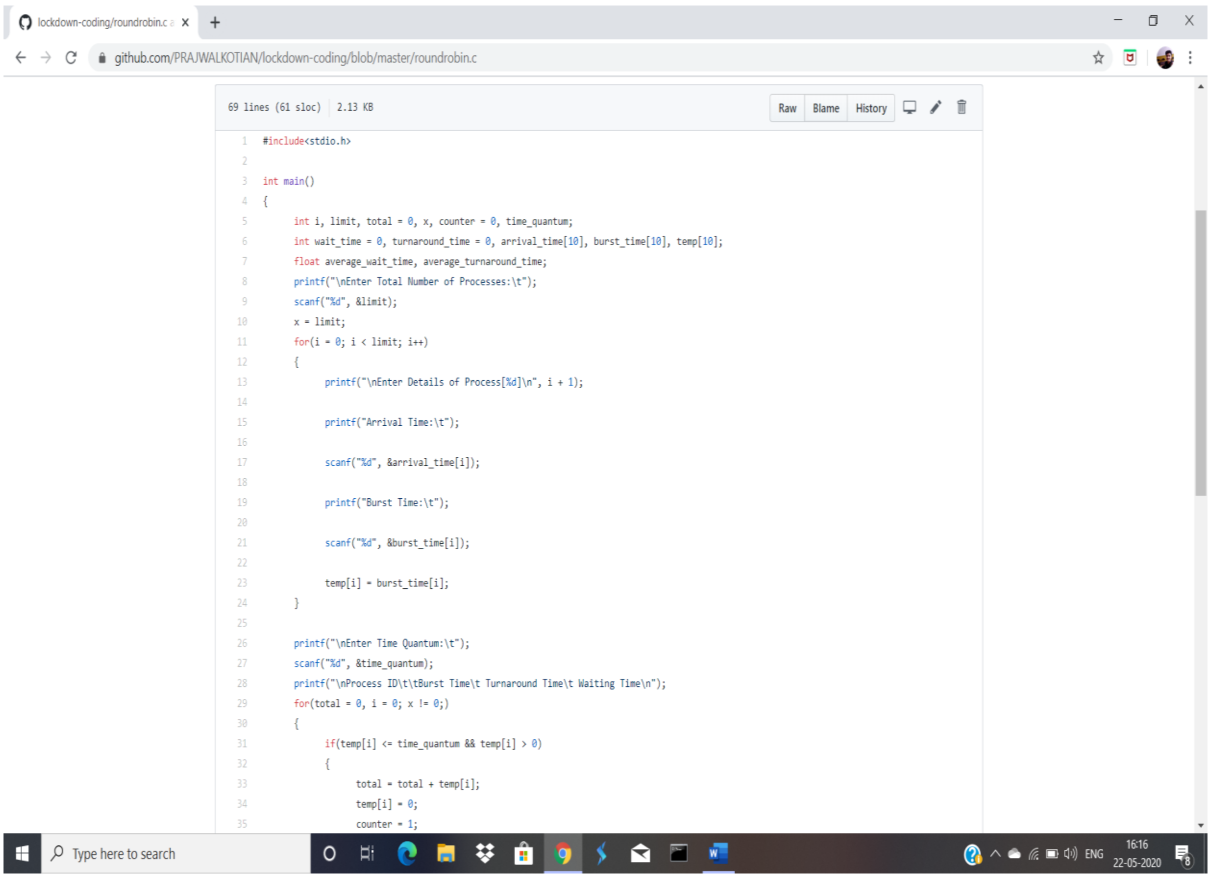
1. Write a C Program to implement various operations on Singly Linked List Stack.



```
#include <stdio.h>
#include <stdlib.h>
void push();
void pop();
void display();
struct node
{
int val;
struct node *next;
};
struct node *head;

void main ()
{
    int choice=0;
    printf("\n*********Stack operations using linked list*********\n");
    printf("\n----------------------------------------------\n");
    while(choice != 4)
    {
        printf("\n\nChose one from the below options...\n");
        printf("\n1.Push\n2.Pop\n3.Show\n4.Exit");
        printf("\n Enter your choice \n");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:
            {
                push();
                break;
            }
            case 2:
            {
                pop();
                break;
            }
            case 3:
```

2. Write a C or Java program to implement round robin type of process scheduling


lockdown-coding/roundrobin.c
github.com/PRAJWALKOTIAN/lockdown-coding/blob/master/roundrobin.c
69 lines (61 sloc) 2.13 KB
Raw
Blame
History
Type here to search
ENG
16:16
22-05-2020

```c
#include<stdio.h>

int main()
{
      int i, limit, total = 0, x, counter = 0, time_quantum;
      int wait_time = 0, turnaround_time = 0, arrival_time[10], burst_time[10], temp[10];
      float average_wait_time, average_turnaround_time;
      printf("\nEnter Total Number of Processes:\t");
      scanf("%d", &limit);
      x = limit;
      for(i = 0; i < limit; i++)
      {
            printf("\nEnter Details of Process[%d]\n", i + 1);

            printf("Arrival Time:\t");

            scanf("%d", &arrival_time[i]);

            printf("Burst Time:\t");

            scanf("%d", &burst_time[i]);

            temp[i] = burst_time[i];
      }

      printf("\nEnter Time Quantum:\t");
      scanf("%d", &time_quantum);
      printf("\nProcess ID\t\tBurst Time\t Turnaround Time\t Waiting Time\n");
      for(total = 0, i = 0; x != 0;)
      {
            if(temp[i] <= time_quantum && temp[i] > 0)
            {
                  total = total + temp[i];
                  temp[i] = 0;
                  counter = 1;
```