

All the best, give your honest attempt

Student Internship Assessment

Instructions:

1. Please DO NOT take any help of any AI tools. We have AI-plagiarism detection tools.
2. Submit your answers in the following format: **Q1_Solution.py** and create **Q1_Writeup.txt** for writeup/explanation/pseudo code.
3. Zip all files in a folder in the following naming convention:
StudentName_CollegeName.zip.
4. Above documents to be uploaded in the following google form:
<https://forms.gle/GUHqfpZncq5LSjx28>

Please write a python program along with brief explanation of the variables used and the correctness of the program. Max score: 60 marks

Q1. [20 marks] Train Station Management Dashboard

Each train station, Durg for example, has many platforms and each incoming train need to be assigned to a platform.

You have to design an object-oriented software solution that:

- a. Keeps track of arrival and departures of trains at each platform
- b. Some trains pass through the platform and will not stop
- c. Allows the station master to change the assignments
- d. Maintain a display board that informs passengers about arrivals (next 1 hr) and departure timings of trains along with their platform numbers

All the best, give your honest attempt

No.	Name	Type	Zone	PF	Departure Days	From	Sch	To	Sch
22647	Korba - Thiruvananthapuram North...	SF	SR	-	S M T W T F S	DURG	00:10	TVCN	15:40
12993	Gandhidham - Puri Weekly SF Expr...	SF	WR	3	S M T W T F S	DURG	00:10	PURI	14:40
22828	Surat - Puri Weekly SF Express (PT)	SF	ECoR	3	S M T W T F S	DURG	00:10	PURI	14:40
20824	Ajmer - Puri SF Express (PT)	SF	ECoR	3	S M T W T F S	DURG	00:10	PURI	14:40
22939	Okha - Bilaspur SF Express (PT)	SF	WR	3	S M T W T F S	DURG	00:10	BSP	03:00
12860	Gitanjali Express (PT)	SF	SER	1	S M T W T F S	DURG	03:45	CSMT	21:20
12870	Howrah - Mumbai CSMT Weekly S...	SF	SER	1,3	S M T W T S	DURG	04:00	CSMT	23:45
22894	Howrah - Sainagar Shirdi SF Express	SF	SER	-	S M T W T F S	DURG	04:00	SNSI	19:10
12768	Santragachi - Hazur Sahib Nanded ...	SF	SCR	-	S M T W T F S	DURG	04:00	NED	18:50
22974	Puri - Gandhidham Weekly SF Expr...	SF	WR	2	S M T W T F S	DURG	04:00	GIMB	05:55
18240	Shivnath Express	Exp	SECR	3	S M T W T F S	DURG	04:10	KRBA	09:15
18110	NSC Bose Itwari - Tatanagar Express	Exp	SER	3	S M T W T F S	DURG	04:40	TATA	20:20
68730	Dongargarh - Raipur MEMU	ME...	SECR	-	S M T W T F S	DURG	04:55	R	06:28
22357	Mumbai LTT - Gaya Weekly SF Expr...	SF	ECR	-	S M T W T S	DURG	05:08	GAYA	22:50
20821	Pune - Santragachi Humsafar Expr...	Hms	SER	3	S M T W T F S	DURG	05:08	SRC	19:00
12808	Samata Express	SF	ECoR	3	S M T W T F S	DURG	05:12	VSKP	16:30
12869	Mumbai CSMT - Howrah Weekly S...	SF	SER	-	S M T W T F S	DURG	05:25	HWH	20:20
22893	Sainagar Shirdi - Howrah SF Express	SF	SER	3	S M T W T F S	DURG	05:25	HWH	20:50
12767	Hazur Sahib Nanded - Santragachi ...	SF	SCR	-	S M T W T F S	DURG	05:25	SRC	19:45
20917	Indore - Puri Humsafar Express	Hms	WR	3	S M T W F S	DURG	05:25	PURI	18:45
20829*	Durg - Visakhapatnam Vande Bharat	VB	SECR	-	S M T W T F S	DURG	05:45	VSKP	13:50
22845	Pune - Hatia SF Express (PT)	SF	SER	3	S M T W T F S	DURG	05:50	HTE	17:25
13426	Surat - Malda Town Express (PT)	Exp	ER	3	S M T W T F S	DURG	06:05	MLDT	06:30
20858	Sainagar Shirdi - Puri Weekly SF Express	SF	ECoR	3	S M T W T F S	DURG	06:05	PURI	20:50
09059	Udhna - Khurda Road Special Fare ...	Exp	WR	-	S M T W T F S	DURG	06:15	KUR	22:30

Q2. [20 marks] Decision Tree is an important machine learning algorithm. You have to develop an Object-oriented implementation of the decision tree. You must implement the following algorithms. Assume that the dataset has only binary attributes and multiple labels.

All the best, give your honest attempt

Algorithm 1 DECISIONTREETRAIN(*data, remaining features*)

```
1: guess  $\leftarrow$  most frequent answer in data // default answer for this data
2: if the labels in data are unambiguous then
3:   return LEAF(guess) // base case: no need to split further
4: else if remaining features is empty then
5:   return LEAF(guess) // base case: cannot split further
6: else // we need to query more features
7:   for all f  $\in$  remaining features do
8:     NO  $\leftarrow$  the subset of data on which f=no
9:     YES  $\leftarrow$  the subset of data on which f=yes
10:    score[f]  $\leftarrow$  # of majority vote answers in NO
11:      + # of majority vote answers in YES
12:      // the accuracy we would get if we only queried on f
13:   end for
14:   f  $\leftarrow$  the feature with maximal score(f)
15:   NO  $\leftarrow$  the subset of data on which f=no
16:   YES  $\leftarrow$  the subset of data on which f=yes
17:   left  $\leftarrow$  DECISIONTREETRAIN(NO, remaining features \ {f})
18:   right  $\leftarrow$  DECISIONTREETRAIN(YES, remaining features \ {f})
19:   return NODE(f, left, right)
20: end if
```

Algorithm 2 DECISIONTREETEST(*tree, test point*)

Q3. [20 marks] Password Generation: Write a function to generate a strong, random password with a mix of uppercase, lowercase, digits, and special characters. User will provide the length of the password.

Now, update the code to accept a password that is easy to remember for the user, but not a strong password. Make changes to it so that it meets the requirements.

