

Topic:

Lesson Substitution

You are required to implement a "Lesson Substitution" program. Assume that teachers are allowed to take leave for a whole day only. When a teacher is on leave, other teachers working in the school may have to take the lessons of the teacher who is on leave. Your program should find appropriate teachers who are available to take his/her lesson(s). Your program should work for any school using 5-day or 6-day cycle and 4 to 8 lessons per day.

For the ABC school which has 3 classes (A, B and C) and employed 5 teachers (RT, SL, TO, UC and VV), uses a 5-day cycle and 4 lessons per day, the teacher timetables are shown below:

RT	Day				
Lesson	1	2	3	4	5
1	A	B	C		A
2		A		B	
3			B	C	B
4	B		A	A	
SL	Day				
Lesson	1	2	3	4	5
1	C			A	
2	A	B	C	C	A
3	B	A	A	B	
4			B		
TO	Day				
Lesson	1	2	3	4	5
1		A		C	
2	C		A		C
3	A	C		A	A
4		C	C		B
UC	Day				
Lesson	1	2	3	4	5
1			A	B	C
2			B	A	B
3		B			
4	A	A		C	A
VV	Day				
Lesson	1	2	3	4	5
1	B	C	B		B
2	B	C			
3	C		C		C
4	C	B		B	C

The timetable of RT can be represented by:

A##B;BA##;C#BA;#BCA;A#B#

The lessons of two days are separated by a ";" and a "#" to denote free lesson.

Each school keeps the teachers' timetables and the number of substituted lessons N of all teachers. At the beginning of each school year, N is zero. If a teacher takes leave and has m lessons on that day, the number of substituted lessons N of that teacher will be deducted by m . The number of substituted lessons of teacher(s) substituted one lesson for that teacher will each be increased by 1.

For example, assume that N of RT is 1 and RT takes leave today which is a Day 1. TO and SL will substitute RT for P1 and P4 respectively. RT's N will be $1 - 2 = -1$. SL's N and TO's N will each be increased by 1.

Each school uses a text file to store all the above data. For example, the file for the ABC school may be:

```
RT;1;A##B;BA##;C#BA;#BCA;A#B#  
SL;-4;CAB#;#BA#;#CAB;ACB#;#A##  
TO;-2;#CA#;A#CC;#A#C;C#A#;#CAB  
UC;3;###A;##BA;AB##;BA#C;CB#A  
VV;2;BBCC;CC#B;B#C#;###B;B#CC
```

The first attribute is the teacher's code. We can see that there are totally five teachers, RT, SL, TO, UC and VV.

The second attribute is N . We can see that RT's N is 1.

The next five attributes show Day 1 to Day 5 lessons. For example, Day 1 of RT is A##B, indicating the first lesson is class A, the second and third lessons are free lessons, and the fourth lesson is class B.

Scanning all data, we can see there are three classes, namely A, B and C. Also, from the data file we know that the school uses a 5-day cycle of 4 lessons per day.

Your program should read the schools' data from a text file SC.TXT of the form

```
001;ABC;5;4  
015;LSC;6;8  
510;EBS;5;7  
...
```

For example, the first line indicates school code of 001 is the ABC which uses a 5-day cycle of 4 lessons per day.

Your program then prompts user to enter the name of the school code. If the school code is valid, it displays the basic information including the school name, the number of days per cycle and the number of lesson per cycle to user. It then requires confirmation from the user. If the answer is no, it asks user to enter the school code again or user can choose to quit.

It then prompts user to enter the location and name of text file storing the timetable. Reading data from this text file, the program checks the number of days per cycle and the number of lessons per day. If anything does not match, it will tell user the problems and the program will terminate.

If match, it will display the codes of all classes and all teachers and asks for confirmation. If not, it will ask the user to check the input file and the program will terminate.

When everything is correct, the program will proceed to find substitution. It asks the user to enter the day of cycle and the code of the teacher who needs substitution. For each lesson needing substitution, it will carry out 4 steps:

Step 1: Find all teachers with a free lesson and therefore available to take that lesson.

Step 2: Among the available teachers, find the teacher(s) with the smallest number of substituted lessons N.

Step 3: Among the teachers found in Step 2, find the teacher(s) with the fewest teaching lessons in his/her timetable on that weekday.

Step 4: Among the teachers found in Step 3, find the teacher whose code is the smallest alphabetically. For example, AA < AB < BB < CC < DA < DZ < EA.

If only one teacher is found after executing a step, that teacher will be suggested to be the substitute teacher.

The program then displays the suggestions and asks for confirmation. The user can accept the substitution or change the substitution by providing a teacher who can substitute.

The above process is repeated for other lesson(s) needing substitution.

Finally, the program will output the substitution form into a text with filename "OUTPUT.TXT", update the data in the data file, and quit. The substitution form can be as simple as:

Name of school: ABC			Substitution for RT on Day 1		
1	A	TO			
4	B	SL			

Apart from the basic features as described above, you may add other useful features to your program. For example, does the program allow a teacher to have more than one substitution in a day? Can the program allow the user to find substitutions for more than one teacher on the same day? Can the program allow the user to find substitutions for teacher(s) on running the program twice?

Useful references

<https://www.w3schools.com/python/>

<https://www.python.org/>

<https://thonny.org/>

Part 1 (Design & Implementation)

Due: Oct 2025 [TBC]

Task 1 Design and Implementation

Describing the design of the solution for the selected problems. Demonstrating results and providing evidence of the implementation process.

Performance	Description
High (17 – 25 Marks)	<ul style="list-style-type: none">• Creates an innovative design for the solution• Demonstrates effective ICT skills in the appropriate use of resources• Demonstrates appropriate ICT skills coherently• Demonstrates effective and comprehensive data/information collection, capture and input• Demonstrates effective and comprehensive data/information verification and validation• Demonstrates effective, efficient and comprehensive data/information processing• Creates an appropriate output format of the deliverables that can easily be tested as a solution to the problem
Middle (9 – 16 Marks)	<ul style="list-style-type: none">• Provides some new ideas in designing the solution• Demonstrates ICT skills in the use of resources with understanding• Demonstrates appropriate ICT skills adequately• Demonstrates reasonable data/information collection, capture and input• Demonstrates reasonable data/information verification and validation• Demonstrates reasonably good quality of data/information processing• Creates an output format of the deliverables that can be tested as a solution to the problem
Low (0 – 8 Marks)	<ul style="list-style-type: none">• Produces a plain design of the solution• Uses resources with relevant ICT skills• Demonstrates appropriate ICT skills occasionally• Demonstrates an awareness of the necessary data/information collection, capture and input• Demonstrates an awareness of the necessary data/information verification and validation• Carries out data/information processing relevant to the problem• Creates an output format of the deliverables that can be partially tested as a solution to the problem

Part 2 (Testing & Evaluation)

Due: Nov 2025 [TBC]

Task 2 Testing and Evaluation

Testing and evaluating the deliverables.

Performance	Description
High (11 – 15 Marks)	<ul style="list-style-type: none">• Records test results comprehensively according to the testing plan with quality metrics• Demonstrates a comprehensive methodology in testing and evaluation• Suggests and explain improvement made as a result of testing• Conducts corrective/preventive actions logically• Evaluates the effectiveness of the deliverables relative to the environmental factors• Presents the findings using a systematic style and appropriate terminology• Organises and present the objectives and original thought effectively• Reflect the student's learning process effectively
Middle (6 – 10 Marks)	<ul style="list-style-type: none">• Records test results generally according to the testing plan• Demonstrates a general methodology in testing and evaluation• Suggests some practical improvement• Conducts some corrective/preventive actions appropriately• Evaluates the effectiveness of the deliverables relative to the partial environmental factors• Presents the findings with a comprehensible and sensible style• Expresses ideas and visions successfully• Reflects the student's learning process
Low (0 – 5 Marks)	<ul style="list-style-type: none">• Records some test results• Demonstrates a basic methodology in testing and evaluation• Attempts to improve the deliverables• Attempts to evaluate the effectiveness of the deliverables• Presents the findings with text and illustration