CS3612 - Intelligent Systems
Assignment: Constraint Satisfaction Problem

Index no: 190290U
Name: P. S. Kalansooriya

Modelling the problem as a Constraint Satisfaction Problem

A Constraint Satisfaction Problem needs to have variable, domains for each variable, and some constraints which determines the value assignment to the variables. Therefore, the given problem can be modelled as a CSP by defining the variables, value that can be assigned to a variable, domains and constraints as follows.

- **Variables** of CSP:
  - the different subjects we get from the input file (eg: Subject_1, Subject_2)

- **A value** that can be assigned to a variable:
  - list consisting of one time slot and a room. eg: [M1, R3]

- **Domain** of variables:
  - The list consisting of each possible value for the variable (subject)

- **Constraints**:
  - A given subjects can be assigned only to one of the possible time slots given for that subject.
  - Two compulsory subjects cannot be in the same time slot (optional subjects may)
  - Two subjects cannot be assigned to the same room if they are assigned

The implementation

1. Got the input and output filenames from the command line arguments

2. Read the input csv file and converted each row to a list:

   Eg: The row `Subject_1, c, M1, M3, Tu2` converted to ➔ `[“Subject_1”, “c”, “M1”, “M3”, “Tu2”]`

3. Read the rooms in the last row of the input file to a list

4. Implemented a dictionary named `subjects_comp_or_opt` to save the subject name as the key and subject status as the value (Compulsory or optional status)

   Eg: `{'Subject_1': 'c', 'Subject_2': 'o', 'Subject_3': 'c', 'Subject_n': 'o'}`

5. Implemented a dictionary named `timeSlots` to save the subject name as the key and the list of available time slots for that subject as the value of an item of the dictionary.

   Eg:

   ```
   {'Subject_1': ['M1', 'M3', 'Tu2'],
   'Subject_2': ['Tu1', 'W1', 'Th2'],
   …}
   ```

6. Implemented a dictionary named `domains` to save the subject name as the key and the all the possible timeslot-room pairs that the subject can be assigned to as the value of an item of the dictionary.

   Eg:

   ```
   {'Subject_1': [['M1', 'R1'], ['M1', 'R2'], ['M1', 'R3'],
   ['M3', 'R1'], ['M3', 'R2'], ['M3', 'R3'], ['Tu2', 'R1'],
   ['Tu2', 'R2'], ['Tu2', 'R3']],
   'Subject_2': [['Tu1', 'R1'], ['Tu1', 'R2'], ['Tu1', 'R3'],
   ['W1', 'R1'], ['W1', 'R2'], ['W1', 'R3'], ['Th2', 'R1'],
   ['Th2', 'R2'], ['Th2', 'R3']],
   …}
   ```

7. Then ran the `backtrackingSearch(subjects_comp_or_opt: Dict, timeSlots: Dict, domains: Dict)` function to get the result.

8. At last, wrote the result to the output csv file.