Review of what we have learned

Computational thinking, algorithm development using natural language, pseudo code, testing and dry-run, IDLE , input, output, data types, mathematical operators, mathematical to python expressions, important functions in math module, strings, indexed access of strings (positive index, negative index, slicing, skipped slicing), some common string related functions along with + and *, mutable and immutable data types, if, if-else, if-elif, if-elif-else, nested-if, while, for, nested-while, nested-for, nested loops and blocks, collection data types (tuple, list, set) their indexed access (positive, negative, slicing, skip-slicing) their functions, collection data types as elements of other collection data types (e.g. list of lists, tuple having list as members etc., local and global variables, , files, functions and parameters, functions with optional parameters, function calls, transfer of function arguments (value, variable, expressions, other functions) to function parameters (positional, named, with optional parameter), text files - reading, writing, appending, CSV files, HTML files,

Today's Topics

Why theory
JSON files and XML files are also Text files
Dictionary data type

https://faculty.math.illinois.edu/~ash/Discrete/213Ch3.pdf

the reachability matrix Mª

If M is the adjacency matrix of a digraph then an entry of 1 in row i, col j indicates an edge vivj, i.e., a path from vi to vj with just one edge. In this section I'll extract from M a new matrix called the reachability matrix, denoted Mª, in which an entry of 1 in row i, col j indicates a path (with one or more edges) from vi to vj, and an entry of 0 means no path at all. In other words, the reachability matrix indicates whether you can get from here to there. (Some books call Mª the transitive closure of M.)

```
begin
   W := M;
   for k = 1 to n do
      for i = 1 to n do
         for j = 1 to n do
            W(i, j) := W(i, j) or (W(i, k) and W(k, j))
end
```

REF:https://slideplayer.com/slide/2389355/9/images/12/Warshall%E2%80%99s+algorithm+%282%29.jpg

JSON files are also text files

Example 1:

```
{
    "fruit": "Apple",
    "size": "Large",
    "color": "Red"
}
```

Example 2:

```
{
    "quiz": {
        "sport": {
            "q1": {
                "question": "Which one is correct team name in
NBA?",
                "options": [
                    "New York Bulls",
                    "Los Angeles Kings",
                    "Golden State Warriros",
                    "Huston Rocket"
                ],
                "answer": "Huston Rocket"
            }
        },
        "maths": {
            "q1": {
                "question": "5 + 7 = ?",
                "options": [
                    "10",
                    "11",
                    "12",
                    "13"
                ],
                "answer": "12"
            },
            "q2": {
                "question": "12 - 8 = ?",
                "options": [
                    "1",
                    "2",
                    "3",
                    "4"
                ],
```

```
                "answer": "4"
            }
        }
    }
}
```

Googlemaps marker JSON data

```
{
  "markers": [
    {
      "name": "Rixos The Palm Dubai",
      "position": [25.1212, 55.1535],
    },
    {
      "name": "Shangri-La Hotel",
      "location": [25.2084, 55.2719]
    },
    {
      "name": "Grand Hyatt",
      "location": [25.2285, 55.3273]
    }
  ]
}
```

XML Files are also text files

Some Linear Programming problems, their formulation and graphical solution:
https://www.superprof.co.uk/resources/academic/maths/linear-algebra/linear-programming/linear-programming-word-problems.html
1Choose the unknowns.
2Write the objective function.
3Write the constraints as a system of inequalities.
4Find the set of feasible solutions that graphically represent the constraints.
5Calculate the coordinates of the vertices from the compound of feasible solutions.
6Calculate the value of the objective function at each of the vertices to determine which of them has the maximum or minimum values. It must be taken into account the possible non-existence of a solution if the compound is not bounded.

http://www0.dlshs.org/webpages/kirknerj/documents/3.3-LinearProgrammingExamples.pdf