Indian Institute of Technology Dharwad

CS 621 Logic and Applications

Assignment 2 : Programming Assignment : Solution
Date of submission: 26 Oct 2023 (5pm)
Mode of submission : email, with subject line – "CS 621: Assignment2"

- Model the puzzle and solve it **using Z3**
- Your submission **must include a ReadMe file**, with instructions on how to execute your code.
- You can choose either Z3cpp or Z3py interfaces.
- You can read the input from file/command line or write the output to a file/command line, however you want to, there are no other restrictions.

---

1. Given an 8 litre bucket of water (say A), and two empty buckets (say B and C) which can contain 5 and 3 litres respectively, you are asked to distribute the water such that there is 4 litres in A , and 4 litres in B.
   What is the minimum number of transfers of water between buckets? Show the stepwise transfers.
   Encode this using satisfiability constraints and solve them using Z3.
   Expected Output:

   ```
   step 0: A:8 B:0 C:0
   transfer from A to C
   step 1: A:5 B:0 C:3
   ...
   step n: A:4 B:4 C:0
   ```

   Answer:

   ```
   Solution found at iteration 8 :
   Step 0 : [8, 0, 0]
   Step 1 : [3, 5, 0]
   Step 2 : [3, 2, 3]
   Step 3 : [6, 2, 0]
   Step 4 : [6, 0, 2]
   Step 5 : [1, 5, 2]
   Step 6 : [1, 4, 3]
   Step 7 : [4, 4, 0]
   ```

   Marking Scheme:
   - Encoding the problem: 5 marks
   - Minimum number of steps:8 (2 mark)
   - Stepwise transfer 3 marks

2. **Definition (Graph colouring)** : Given Graph $G = (V, E)$ a valid colouring of a graph is an assignment of colours to the vertices of the graph so that no two adjacent vertices have the same colour.
   Write a program that can assign colours to each vertex in G.
   Input:
   - A set of vertices $V = [A, B, C, D]$
   - Pairs of edges $E = [(A, B), (A, C), (A, D), (B, D), (C, D)]$

   Expected Output:
   - Total number of colours required in a valid colouring of $G : 3$
   - A valid assignment of colours to all vertices in V (say) $A : 0, B : 1, C : 1, D : 2$

   The naive approach would be to assign a different colour to each vertex, which is not interesting. This approach will not be accepted as a valid solution. One should give the *least possible number* of colors possible.
   Marking Scheme:
   - Encoding the graph colouring problem (5 marks)
   - Valid assignment of colours for test inputs(4 marks)
   - total number of colours (1 mark)

3. **Scheduling Problem** An employer needs to interview N candidates, and therefore makes N interview slots. Every person has a free-busy schedule for those slots.
   (a) Can you schedule the interview, such that there is exactly one candidate allocated for a timeslot, for each of the N candidates?
   (b) How many such schedules can you make?
   - **Input:** The input to your program, can be a binary matrix of this form, where columns represent candidates $C_0, \cdots, C_{N-1}$ and rows represent time slots $T_0, \cdots, T_{N-1}$.
     If a candidate is free, it is represented by 1. If the candidate is busy, it is represented by 0.

	C0	C1	C2
TS0	1	0	0
TS1	0	0	0
TS2	1	1	1

     In the above figure, in timeslot T_0, only candidate C_0 is free; in timeslot TS_2, all candidates are busy.
   - **Expected Output:**
     (a) For (a), if satisfiable, display the interview schedules
     (b) For (b) Count how many such schedules are possible? Print all possible solutions.

   Note: Your program has to work for any given $N \times N$ binary matrix as input.
   Answer:

```
Input:
1 0 0 0 0
1 1 0 0 0
1 0 0 1 0
0 1 1 1 1
0 1 1 1 0

Expected Output:
Solution 1 [0, 1, 3, 2, 4]
Solution 2 [0, 1, 3, 4, 2]
Number of solutions is 2
```

Marking Scheme:
- The program is expected to work for any input $N \times N$ binary matrix, if not, marks will be awarded accordingly.
- Encoding the problem: 5 marks
- Display schedules 4 marks
- Count how many solutions are possible 1 mark
- The program is checked with several test cases, if it fails, then marks will be awarded accordingly.