Project 3

Cam Hayes

The system works by initially creating a dictionary of attributes and their binary representation, encoding the attributes into a list composed of ordered lists for each object, a constraints set defined by the constraints file and the feasibility set generated by iterating over all of the encoded objects and checking them against the constraints. Because pySAT uses a binary integer representation for objects, the system maintains that structure in all its data objects and references a find_in_dict method when a textual representation is needed.

After generating the workable data sets, the system engages the menu interfaces. Because of time/scope constraints, there is a functional method for penalty and qualitative logic for each of the main data processing components.

Penalty Logic:
A penalty logic set is generated through the provided preference file. This creates a binary integer PySAT friendly interpretation of each line in the penalty logic file. The logic follows that a formula is split into clauses at AND, and their literals are evaluated at OR and NOT. An example output of a rule would be:

> beef = -3, cake = 1; soup = -2
>
> Rule: beef OR cake AND NOT soup
>
> Output: [[-3, 1], [2]]

Processing penalty logic involves iterating over every rule for every feasible object. If a rule is solved by an object, then the penalty appended to the data structure for that object is 0. If it does violate the rule, the actual penalty is appended to the data structure for that object. Processing the penalty logic returns a list of processed object sets, where the 0th index is the object number and the remaining members are penalty costs incurred for rules n, n+1, … etc.

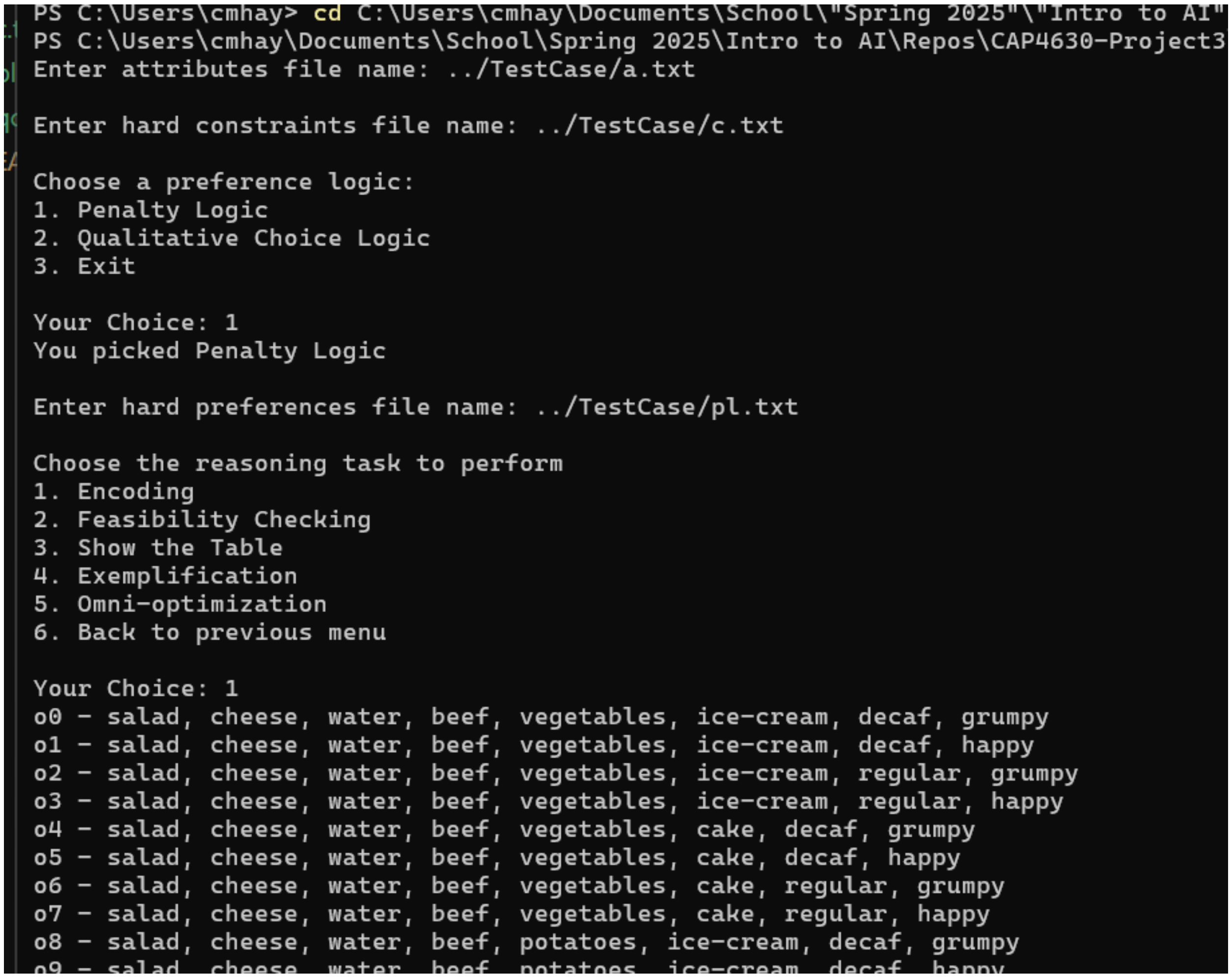
Exemplifying and optimizing penalty logic is a simple comparison of the sum of incurred penalties between each object.

Qualitative Choice Logic:

A similar logic set is generated for QCL, with the exception of a BT and IF delimiter. The object set for each QCL rule is a composite of the rules themselves, followed by the initial condition which is appended to the end of the list. Processing QCL occurs by first processing whether the initial condition is applicable. If so, each condition is processed until the first successful solution. If there is no solution or if the initial condition failed, the infinity value is appended to the object. The object set for processed QCL objects is formatted such that each member of the list if the preference found during evaluation. For optimization and exemplification of QCL sets, the program iterates over the vector components for two vectors, and determines whether or not comparison rules in Pareto optimization apply at each vector component. This is, for example, if x_1 > y_2, x_2 < y_2, x_3 == y_3, x_4 OR y_4 == INFINITY, etc. The result of that comparison is returned and evaluated by the calling method.

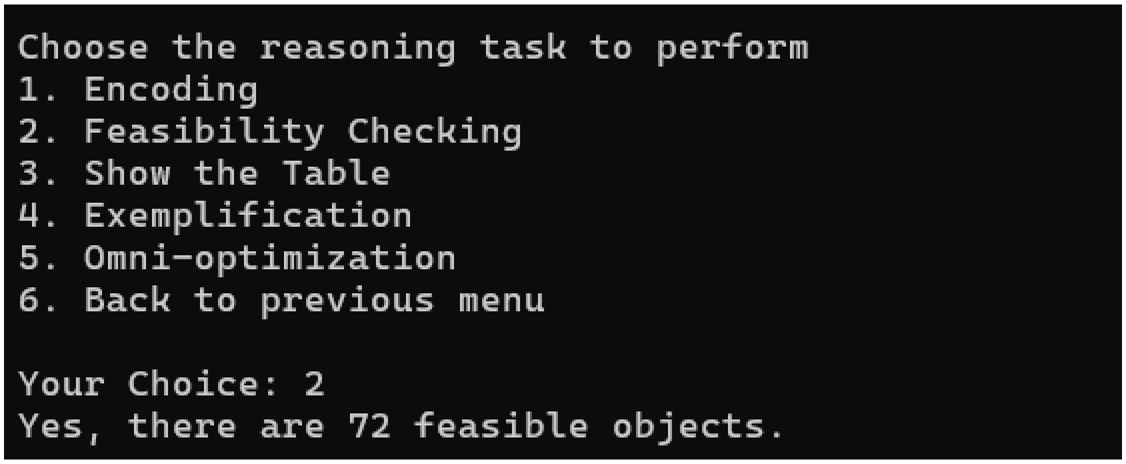
The remaining methods handle basic dispatch and utility services.

Initialization and Encoding Print:



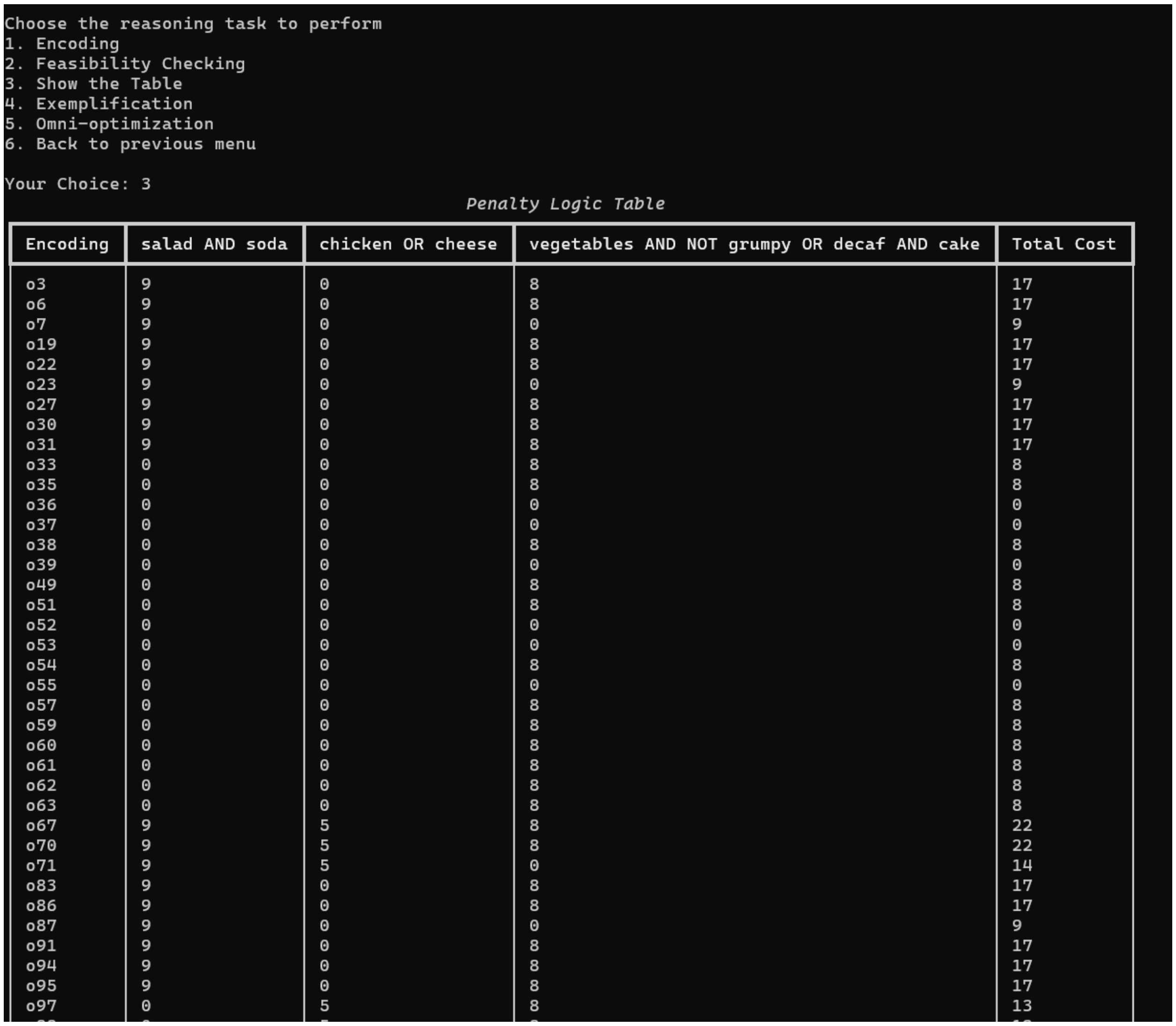
```
PS C:\Users\cmhay> cd C:\Users\cmhay\Documents\School\"Spring 2025"\"Intro to AI"
PS C:\Users\cmhay\Documents\School\Spring 2025\Intro to AI\Repos\CAP4630-Project3
Enter attributes file name: ../TestCase/a.txt

Enter hard constraints file name: ../TestCase/c.txt

Choose a preference logic:
1. Penalty Logic
2. Qualitative Choice Logic
3. Exit

Your Choice: 1
You picked Penalty Logic

Enter hard preferences file name: ../TestCase/pl.txt

Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 1
o0 - salad, cheese, water, beef, vegetables, ice-cream, decaf, grumpy
o1 - salad, cheese, water, beef, vegetables, ice-cream, decaf, happy
o2 - salad, cheese, water, beef, vegetables, ice-cream, regular, grumpy
o3 - salad, cheese, water, beef, vegetables, ice-cream, regular, happy
o4 - salad, cheese, water, beef, vegetables, cake, decaf, grumpy
o5 - salad, cheese, water, beef, vegetables, cake, decaf, happy
o6 - salad, cheese, water, beef, vegetables, cake, regular, grumpy
o7 - salad, cheese, water, beef, vegetables, cake, regular, happy
o8 - salad, cheese, water, beef, potatoes, ice-cream, decaf, grumpy
o9 - salad, cheese, water, beef, potatoes, ice-cream, decaf, happy
```

Feasibility Display:



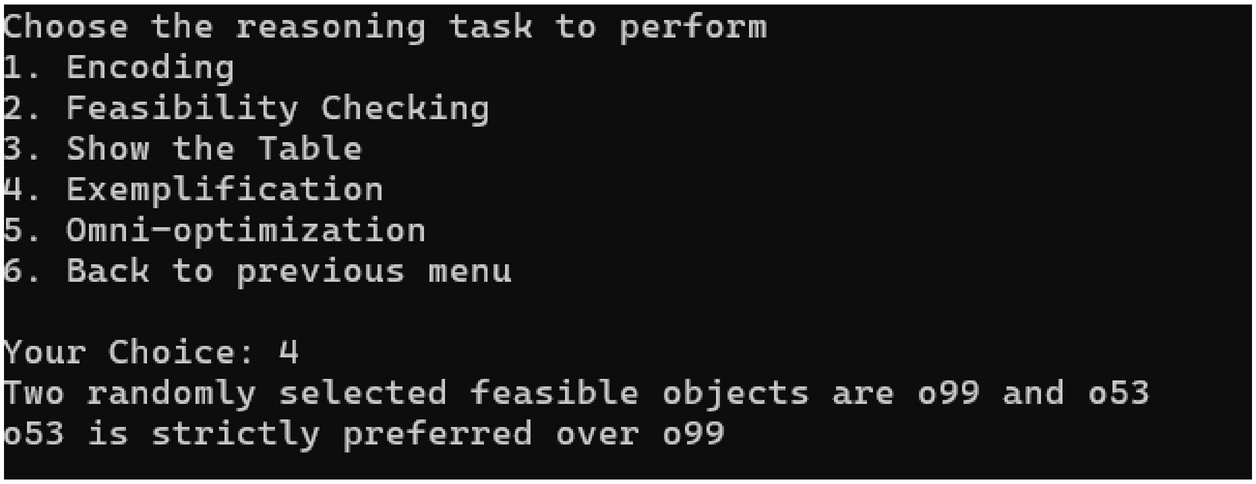
```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 2
Yes, there are 72 feasible objects.
```

Penalty Logic Table:



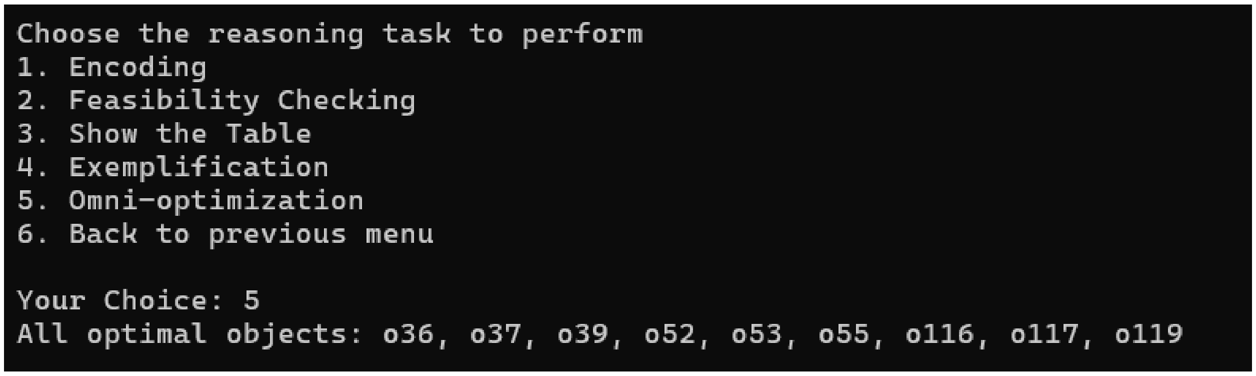
```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 3
                                Penalty Logic Table
 Encoding   salad AND soda   chicken OR cheese   vegetables AND NOT grumpy OR decaf AND cake   Total Cost

 o3         9                0                   8                                             17
 o6         9                0                   8                                             17
 o7         9                0                   0                                             9
 o19        9                0                   8                                             17
 o22        9                0                   8                                             17
 o23        9                0                   0                                             9
 o27        9                0                   8                                             17
 o30        9                0                   8                                             17
 o31        9                0                   8                                             17
 o33        0                0                   8                                             8
 o35        0                0                   8                                             8
 o36        0                0                   0                                             0
 o37        0                0                   0                                             0
 o38        0                0                   8                                             8
 o39        0                0                   0                                             0
 o49        0                0                   8                                             8
 o51        0                0                   8                                             8
 o52        0                0                   0                                             0
 o53        0                0                   0                                             0
 o54        0                0                   8                                             8
 o55        0                0                   0                                             0
 o57        0                0                   8                                             8
 o59        0                0                   8                                             8
 o60        0                0                   8                                             8
 o61        0                0                   8                                             8
 o62        0                0                   8                                             8
 o63        0                0                   8                                             8
 o67        9                5                   8                                             22
 o70        9                5                   8                                             22
 o71        9                5                   0                                             14
 o83        9                0                   8                                             17
 o86        9                0                   8                                             17
 o87        9                0                   0                                             9
 o91        9                0                   8                                             17
 o94        9                0                   8                                             17
 o95        9                0                   8                                             17
 o97        0                5                   8                                             13
```

Exemplification (penalty logic):



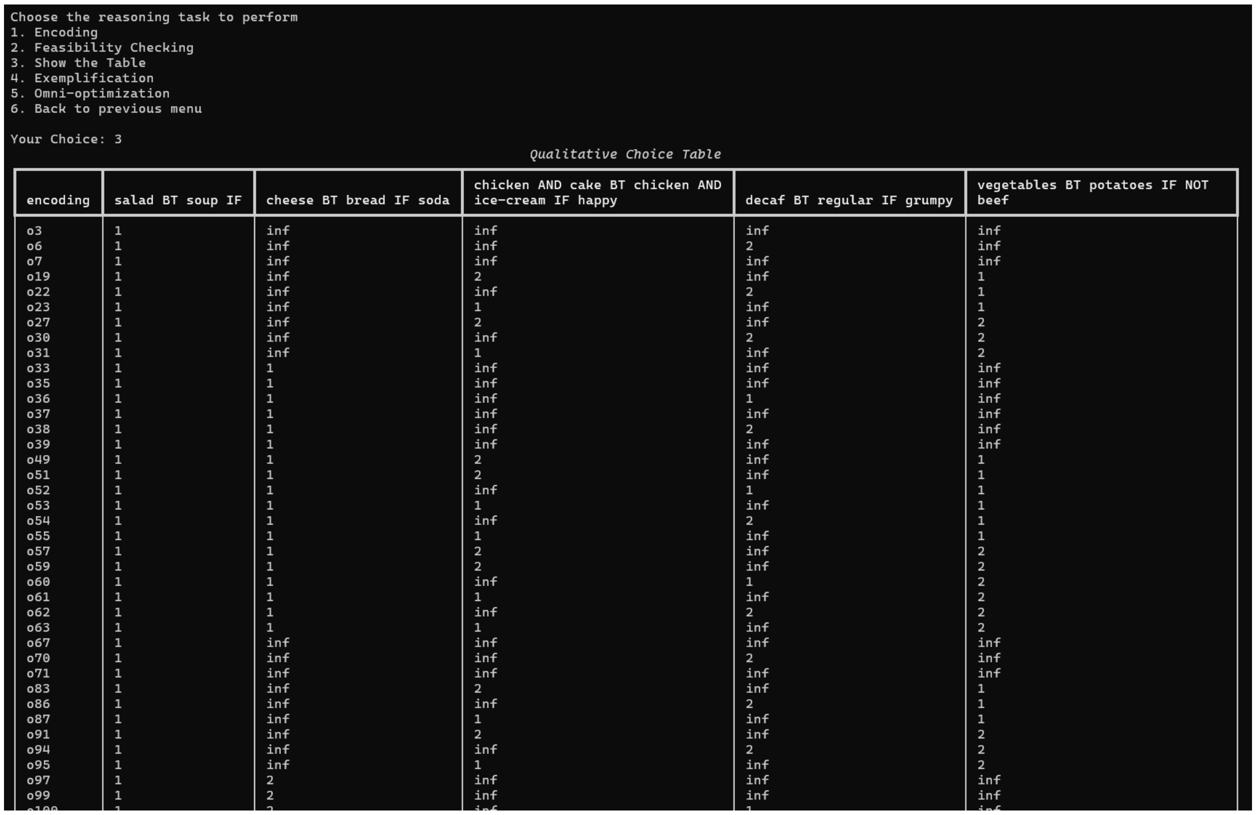
```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 4
Two randomly selected feasible objects are o99 and o53
o53 is strictly preferred over o99
```

Omni-optimization (penalty logic):



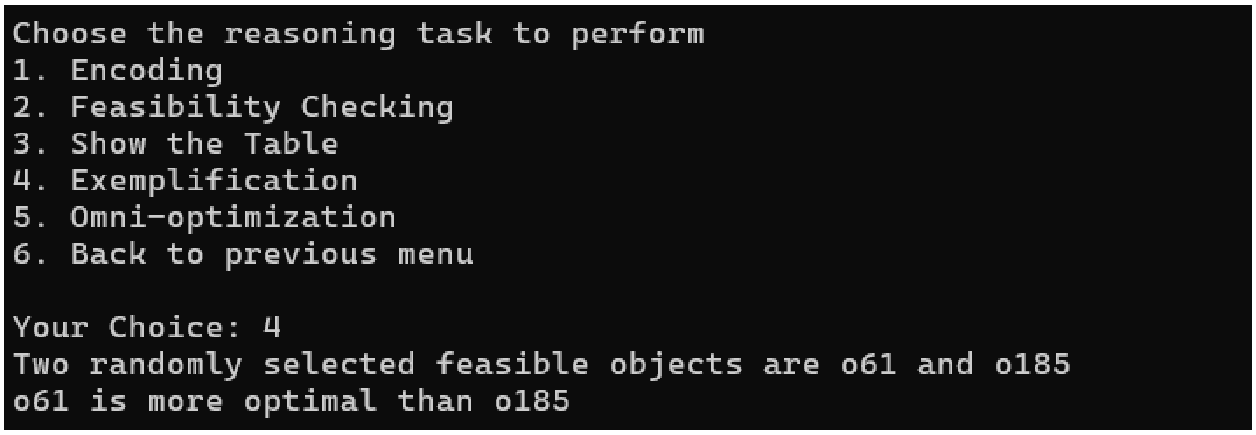
```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 5
All optimal objects: o36, o37, o39, o52, o53, o55, o116, o117, o119
```

Qualitative Choice Table:



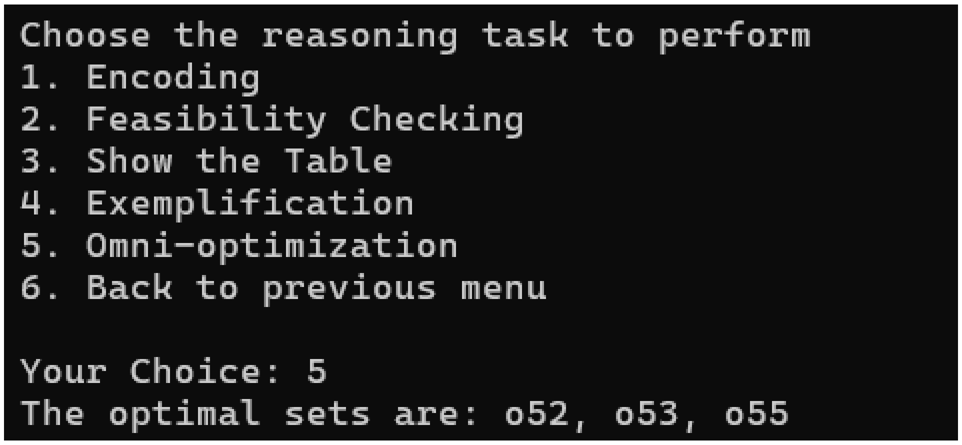
```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 3
                                          Qualitative Choice Table
┌──────────┬──────────────────┬─────────────────────────┬─────────────────────────────────┬────────────────────────────┬─────────────────────────────────┐
│ encoding │ salad BT soup IF │ cheese BT bread IF soda │ chicken AND cake BT chicken AND │ decaf BT regular IF grumpy │ vegetables BT potatoes IF NOT   │
│          │                  │                         │ ice-cream IF happy              │                            │ beef                            │
├──────────┼──────────────────┼─────────────────────────┼─────────────────────────────────┼────────────────────────────┼─────────────────────────────────┤
│ o3       │ 1                │ inf                     │ inf                             │ inf                        │ inf                             │
│ o6       │ 1                │ inf                     │ inf                             │ 2                          │ inf                             │
│ o7       │ 1                │ inf                     │ inf                             │ inf                        │ inf                             │
│ o19      │ 1                │ inf                     │ 2                               │ inf                        │ 1                               │
│ o22      │ 1                │ inf                     │ inf                             │ 2                          │ 1                               │
│ o23      │ 1                │ inf                     │ 1                               │ inf                        │ 1                               │
│ o27      │ 1                │ inf                     │ 2                               │ inf                        │ 2                               │
│ o30      │ 1                │ inf                     │ inf                             │ 2                          │ 2                               │
│ o31      │ 1                │ inf                     │ 1                               │ inf                        │ 2                               │
│ o33      │ 1                │ 1                       │ inf                             │ inf                        │ inf                             │
│ o35      │ 1                │ 1                       │ inf                             │ inf                        │ inf                             │
│ o36      │ 1                │ 1                       │ inf                             │ 1                          │ inf                             │
│ o37      │ 1                │ 1                       │ inf                             │ inf                        │ inf                             │
│ o38      │ 1                │ 1                       │ inf                             │ 2                          │ inf                             │
│ o39      │ 1                │ 1                       │ inf                             │ inf                        │ inf                             │
│ o49      │ 1                │ 1                       │ 2                               │ inf                        │ 1                               │
│ o51      │ 1                │ 1                       │ 2                               │ inf                        │ 1                               │
│ o52      │ 1                │ 1                       │ inf                             │ 1                          │ 1                               │
│ o53      │ 1                │ 1                       │ 1                               │ inf                        │ 1                               │
│ o54      │ 1                │ 1                       │ inf                             │ 2                          │ 1                               │
│ o55      │ 1                │ 1                       │ 1                               │ inf                        │ 1                               │
│ o57      │ 1                │ 1                       │ 2                               │ inf                        │ 2                               │
│ o59      │ 1                │ 1                       │ 2                               │ inf                        │ 2                               │
│ o60      │ 1                │ 1                       │ inf                             │ 1                          │ 2                               │
│ o61      │ 1                │ 1                       │ 1                               │ inf                        │ 2                               │
│ o62      │ 1                │ 1                       │ inf                             │ 2                          │ 2                               │
│ o63      │ 1                │ 1                       │ 1                               │ inf                        │ 2                               │
│ o67      │ 1                │ inf                     │ inf                             │ inf                        │ inf                             │
│ o70      │ 1                │ inf                     │ inf                             │ 2                          │ inf                             │
│ o71      │ 1                │ inf                     │ inf                             │ inf                        │ inf                             │
│ o83      │ 1                │ inf                     │ 2                               │ inf                        │ 1                               │
│ o86      │ 1                │ inf                     │ inf                             │ 2                          │ 1                               │
│ o87      │ 1                │ inf                     │ 1                               │ inf                        │ 1                               │
│ o91      │ 1                │ inf                     │ 2                               │ inf                        │ 2                               │
│ o94      │ 1                │ inf                     │ inf                             │ 2                          │ 2                               │
│ o95      │ 1                │ inf                     │ 1                               │ inf                        │ 2                               │
│ o97      │ 1                │ 2                       │ inf                             │ inf                        │ inf                             │
│ o99      │ 1                │ 2                       │ inf                             │ inf                        │ inf                             │
│ o100     │ 1                │ 2                       │ inf                             │ 1                          │ inf                             │
```

Exemplification (Qualitative Choice):



```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 4
Two randomly selected feasible objects are o61 and o185
o61 is more optimal than o185
```

Omni Optimization (Qualitative Choice):



```
Choose the reasoning task to perform
1. Encoding
2. Feasibility Checking
3. Show the Table
4. Exemplification
5. Omni-optimization
6. Back to previous menu

Your Choice: 5
The optimal sets are: o52, o53, o55
```