Discrete Optimization Specialization: Workshop 3
Beauty Trap: Part 1

1 Problem Statement

Wang Yun begins a plot to help defeat Dong Zhou. He sets out to have both Dong Zhou and his best fighter Lü Bu fall for his foster daughter Diao Chan, who is a remarkable beauty. When Dong Zhou is invited to Wang Yun's house Diao Chan prepares a magnificient banquet in order to seduce him. This feast trap is the first part of the "beauty trap".

Diao Chan must plan the most magnificent feast possible. The sequence of dishes must be exquisite and balanced so that the whole sequence is a magnificient meal.

Diao Chan knows a number of dishes she can make. She can only serve each dish once, obviously.

Each dish tastes spicy, sour, salty, sweet, umami or bland. She should not serve two dishes of the same taste in a row. The first dish should be salty, and the last dish should be sweet. After a spicy dish the next dish must be bland or sweet. After a sour dish the next dish must be bland or umami. No spicy or umami dishes directly after a sweet dish.

Each dish is served hot, cold or warm. Between serving a hot dish and a later cold dish there must be a warm dish.

Each dish is either heavy on the stomach or light. Diao Chan's banquet cannot have more than 2 heavy dishes in a row.

Her aim is to maximize the magnificience of the meal, which is given by the sum of the value of the dishes, plus the number of changes in taste, temperature and "weight" between each pair of consecutive dishes.

2 Data Format Specification

The input form for the Feast Trap is a file named `data/feasttrap_*.dzn`, where p is the problem number with len is the number of courses, $DISH$ is a set of dishes, $taste$ is an array mapping dishes to taste, $temp$ is an array mapping dishes to temperature, $heavy$ is an array of Booleans showing which dishes are heavy, $value$ is an array mapping dishes to their value in magnificience. The decisions are $dish$ which is an array of length len showing which dish is offered in each course.

The data declarations and decisions are hence:

```
enum DISH;
enum TASTE = {spicy, sour, salty, sweet, umami, bland};
enum TEMP = {hot, cold, warm};

array[DISH] of TASTE: taste;
array[DISH] of TEMP: temp;
array[DISH] of bool: heavy;
array[DISH] of int: value;

int: len; % length of banquet
set of int: COURSE = 1..len;
```

```
array[COURSE] of var DISH: dish;
```

An example data file is

```
len  = 6;

DISH =  { MAPOTOFU, KUNGPAOCHICKEN, COCONUTJELLY, LAIWONGBAO, CHARSIUBAO,
          SESAMEPRAWN, HOTSOURSOUP, CHILIDUMPLINGS, GLASSNOODLES, FRIEDRICE };

taste = [umami, spicy, sweet, sweet, salty, salty, sour, spicy, bland, bland];
temp =  [ hot, hot, warm, cold, warm, warm, hot, hot, cold, warm];
heavy = [ true, true, false, false, false, true, true, true, false, false];
value = [ 8, 4, 5, 3, 5, 7, 4, 3, 2, 1 ];
```

which considers a 6 course banquet chosen from 10 possible dishes. Your model should contain a decision variable dish which decides what is served for each course, and an objective variable obj defining the objective. For example it might output

```
dish = [CHARSIUBAO, LAIWONGBAO, GLASSNOODLES, SESAMEPRAWN, MAPOTOFU, COCONUTJELLY];
obj = 41;
```

3   Technical Requirements

For completing the workshop you will need MINIZINC 2.2.x (`http://www.minizinc.org/software.html`).

3

hot

1 : hot

hot

warm

2 :   cold
warm

cold,warm