The Chinese University of Hong Kong

Weeks 5 and 6

Transforming Data

- Very often, the given data to a problem represent only a certain viewpoint
- We need to transform the data to allow easy expressions of constraints in the problem
- There are two ways to transform data
  - Write an external program to transform the data and create a new .dzn file
  - Use the give .dzn file and transform data by constraints

In-class Survey

- The disadvantage of transforming data by using an outside program to add it to the data file is
  - A: the transformed data may not agree with original data
  - B: it is more difficult to write
  - C: it solves less efficiently
  - D: it makes the model larger
  - E: NONE of the above

In-class Survey

- The advantage of transforming data by using constraints is
  - A: the transformed data agrees with original data
  - B: it is typically easier to write
  - C: it solves more efficiently
  - D: it makes the model smaller
  - E: NONE of the above

Data Transformation Exercise 1

- Given a set PERSON, you are given likes data of the form, where `like[p1]` is the set of people that `p1` likes

```
array[PERSON] of set of PERSON: like;
```

- Build an array `l[PERSON,PERSON]` such that `l[p1,p2]` is true if `p1` likes `p2`

```
array[PERSON,PERSON] of bool: l;
```

Data Transformation Exercise 2

- Given a set PERSON, you are given likes data of the form an array `l[PERSON,PERSON]` such that `l[p1,p2]` is true if `p1` likes `p2`

```
array[PERSON,PERSON] of bool: l;
```

- Build an array `like[PERSON]` of sets of person such that `like[p1]` is the set of people liked by `p1`

```
array[PERSON] of set of PERSON: like;
```

The Chinese University of Hong Kong

Let's look at Survey 5

In-class Survey

- What is the value of h in the following?

```
var 0..100: x = y + 2;
var 4..7: y;
int: h = lb(x);
```

  - A: 0
  - B: 4
  - C: 6
  - D: 2
  - E: any of the above

Tracing

- What is the output of the following?

```
var 0..6: x = y + 2;
var 0..6: y;
array[1..5] of int: a = [6,2,3,1,4];
constraint forall(i in lb(x)..ub(x))
                (trace("\(a[i]) ",
                       y != a[i]));
```

Tracing

- What is the output of the following?

```
var 0..6: x = y + 2;
var 0..6: y;
array[1..5] of int: a = [6,2,3,1,4];
constraint forall(i in lb(x)..ub(x))
                (trace("\(a[i]) ",
                       y != a[i]));
```

- 2 3 1 4 … array access out of bounds!

Tracing

- What is the output of the following?

```
var 0..6: x = y + 2;
var 0..6: y;
array[1..5] of int: a = [6,2,3,1,4];
constraint forall(i in lb(x)..ub(x))
                (trace("\(a[i]) ",
                       y != a[i]));
```

- 2 3 1 4 … array access out of bounds!
- This is not a safe use of lb(x) and ub(x)!

The Chinese University of Hong Kong

Model Debugging

- Download
  - freepizza.mzn + freepizza*.dzn
  - from the course homepage
- Generate the best solution you can for the data by hand
- Correct the freepizza model

The Chinese University of Hong Kong

Let's look at Survey 6

Interesting Dinner Party

- An interesting dinner party for n people around a circular table requires
  - each couple is seated either adjacent or far apart $> n$ div 3 positions
  - and the interest of the dinner is the number of rivals which are seated adjacent or far apart.
- BUILD a MiniZinc model!
- Predicates help!

Interesting Dinner Party Data

```
int: n;
int: m;
array[1..m,1..2] of 1..n: couple;
int: k
array[1..k,1..2] of 1..n: rival;
```

Interesting Dinner Party Data

- Make a maximally interesting dinner party for the data by hand

```
n = 8;
m = 3;
couple = [| 1,2 | 3,4 | 5,6 |];
k = 6;
rival = [| 1,3 | 1,6 | 2,8 |
           4,6 | 4,7 | 7,8 |];
```

In-class Survey

- What is the context of "`x <-> y`" in the following?

```
constraint b -> ((u + (x <-> y)) >= 0);
```

where `b`, `x`, `y` are Bools and `u` an int

  - A: root
  - B: positive
  - C: negative
  - D: mixed
  - E: NONE of the above

More Context

- What are the contexts for all terms in the following constraint?

```
constraint y > bool2int(b) ->
           (a[z] -> x = 5);
```

where `y`, `x` are `int` and `b`, `a[]`, `z` are `Bool`

y > bool2int(b)	b
a[z]	z
x = 5	x

More Context

- What are the contexts for all terms in the following constraint?

```
constraint y > bool2int(b) ->
           (a[z] -> x = 5);
```

where `y`, `x` are `int` and `b`, `a[]`, `z` are `Bool`

- y > bool2int(b) — - b — ±
- a[z] — - z — ±
- x = 5 — + x — +

Local Variables

- Modify the following predicate definition so that it can be used in any context

```
predicate p(var int: x, int: n) =
    let { var 0..n: y; } in
    x + y = n /\ y*y - x*x <= n;
```