FIT5216: Modelling Discrete Optimization Problems
Inclass Task 4: Team Select

1 Problem Statement

Each captain Xavier, Yuri, and Zena has to pick an indoor soccer teams from the players:

- **Goalies**: Ant, Bee
- **Defence**: Chu, Deb, Eve, Fin
- **Offence**: Ged, Hel, Ila, Jan, Kim

Each team has to have one goalie, two defence, and two offence and one reserve (of any type). The teams of Xavier and Yuri can only have two common members, similarly for Xavier and Zena. Each captain has a perceived value of each player. Choose the teams which maximise the total perceived value.

The percieved values are given by

```
enum PLAYER = {Ant, Bee, Chu, Deb, Eve, Fin, Ged, Hel, Ila, Jan, Kim};
enum CAPTAIN = {Xavier, Yuri, Zena};
array[CAPTAIN,PLAYER] of int: value =
   [| 2, 5, 6, 8, 9, 5, 8, 7, 7, 4, 6
    | 9, 8, 4, 7, 6, 4, 5, 3, 5, 5, 7
    | 8, 4, 3, 3, 6, 2, 5, 5, 3, 2, 5 |];
```

Build a MiniZinc model `teamselect_set.mzn` which solves the problem. It should use the decision variables and output statement:

```
var set of PLAYER: xavier;
var set of PLAYER: yuri;
var set of PLAYER: zena;
output ["xavier = \(xavier);\nyuri = \(yuri);\nzena = \(zena);\n"];
```

Build a new MiniZinc model `teamselect_array.mzn` which solves the problem It should use the decision variables and output:

```
set of int: POS = 1..6;
array[CAPTAIN,POS] of var PLAYER: team;
output ["team = array2d(CAPTAIN,POS,\(team));\n"];
```

2 Instructions

Edit the provided `mzn` model files to solve the problems described above. Your implementations can be tested locally by using the *Run* icon in the MiniZinc IDE or by using,

```
minizinc ./modelname.mzn
```

at the command line.