FIT5216: Modelling Discrete Optimization Problems
Inclass Task 7: SearchParty

1 Problem Statement

Given 6 searchers `A,B,C,D,E,F` each of which must search in a direction `n,e,s,w`. The following constraint apply to the search:

- Each direction must have at least 1 searcher.
- If `A` searches `n` or `s` then `B` cannot search `n` or `s`.
- `C` and `D` must search in opposite directions.
- If either `E` or `F` go `w` then no one else can search that direction.
- The pairs (A,E), (B,F), (C,E) can't search the same direction.

Build a MiniZinc model to find a solution. Use the variable declarations:

```
enum DIRN = {n,e,s,w};
var DIRN: A;
var DIRN: B;
var DIRN: C;
var DIRN: D;
var DIRN: E;
var DIRN: F;
```

Make sure you dont add an output statement (so the default output is used!).

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.