Challenge June 2024
Stable marriages

A solution with OPL CPLEX CPOptimizer by Alex Fleischer
afleischer@fr.ibm.com

In January 2024, the challenge was a family riddle and I am glad to help families again here with optimization and CPLEX. Yes, CPLEX and optimization can help us make a better world.

Within CPLEX, I used OPL and chose the Constraint Programming engine: CPOptimizer.

.mod

```
using CP;


{string} men=...;
{string} women=...;

range r=1..card(men);


int menRankWomen[r][r]=...;
int womenRankMen[r][r]=...;

dvar int wife[r] in r;
dvar int husband[r] in r;

subject to
{
  // all men and women have different spouses and wife is the inverse of husband
  allDifferent(wife);
  allDifferent(husband);
  inverse(wife,husband);

  // stability equations

  forall(m in r) forall(w in r)
     ((w!=wife[m]) && (menRankWomen[m][w]<=menRankWomen[m][wife[m]]))
     =>
     (womenRankMen[w][m]>=womenRankMen[w][husband[w]]);

   forall(w in r) forall(m in r)
     ((m!=husband[w]) && (womenRankMen[w][m]<=womenRankMen[w][husband[w]]))
     =>
     (menRankWomen[m][w]>=menRankWomen[m][wife[m]]);
```

```
}

execute display_solution
{
  writeln("Stable marriages");
  for(var m in men) writeln(m," - ",Opl.item(women,wife[1+Opl.ord(men,m)]-1));
}

/* One solution is

Stable marriages
Adam - Barbara
Bob - Doris
Charlie - Claire
Dave - Alice
Edgar - Elsie
*/

// In order to enumerate uncomment the following lines

//main
//{
//      var nb=0;
//    cp.param.SearchType=24;
//    cp.param.workers=1;
//
//    thisOplModel.generate();
//    cp.startNewSearch();
//    while
//    (cp.next()) {  thisOplModel.postProcess(); nb++; writeln("nb=",nb);}
//}
```

.dat

```
men={"Adam","Bob","Charlie","Dave","Edgar"};
women={"Alice","Barbara","Claire","Doris","Elsie"};

menRankWomen=
[[5,1,2,4,3],
[4,1,3,2,5],
[5,3,2,4,1],
[1,5,4,3,2],
[4,3,2,1,5]];

womenRankMen=
[[1,2,4,3,5],
[3,5,1,2,4],
[5,4,2,1,3],
[1,4,3,2,5],
[4,2,3,5,1]];
```

And If I uncomment the enumerate part I can prove that there are only five solutions.