```
/*********************************************
 * OPL 12.8.0.0 Model
 * Author: chirs + kipp
 * Creation Date: 19 Mar 2019 at 5:03:48 pm
 *********************************************/

range   Week = 1..54;
range   Intern = 1..11;
range   Rotation = 1..18;


dvar boolean    x[Intern][Rotation][Week];


dvar boolean    y1[Intern][Week];
dvar boolean    y2[Intern][Week];
dvar boolean    y3[Intern][Week];
dvar boolean    y4[Intern][Week];
dvar boolean    y5[Intern][Week];
dvar boolean    y6[Intern][Week];
dvar boolean    y7[Intern][Week];
dvar boolean    y8[Intern][Week];
dvar boolean    y9[Intern][Week];
dvar boolean    y10[Intern][Week];
dvar boolean    y11[Intern][Week];
dvar boolean    y4_1[Intern][Week];

dvar boolean    L1[Week];

int             M = 1000;

dexpr int   z = sum(i in Intern, j in Rotation, k in Week) x[i][j][k];

maximize z;


subject to{

//Rotation 17 will be for the extra MIC rotation
//let Rotation 18 be Weeks 1-4 when Interns 7-11 have not yet begun and
// Weeks 51-54 when Intern 1-6 have finished

forall(i in 1..6)
  sum(k in 51..54)x[i][18][k] ==4;

forall(i in 7..11)
  sum(k in 1..4)x[i][18][k] ==4;

forall(i in 1..6)
  sum(k in 1..50)x[i][18][k] ==0;

forall(i in 7..11)
  sum(k in 5..54)x[i][18][k] ==0;

//Orientation Constraint
//forall(i in 1..6)
//  sum(k in 1..4)x[i][5][k] ==1;
```

```
58
59 //forall(i in 7..11)
60 //  sum(k in 5..8)x[i][5][k] ==1;
61
62 //forall(i in 1..6)
63 //  sum(k in 1..4)x[i][9][k] ==1;
64
65 //forall(i in 7..11)
66 //  sum(k in 5..8)x[i][9][k] ==1;
67
68 //forall(i in 1..6)
69 //  sum(k in 1..4)x[i][10][k] ==2;
70
71 //forall(i in 7..11)
72 //  sum(k in 5..8)x[i][10][k] ==2;
73
74 ///Intern Physical Constraint (can only be in one place at a time)
75 forall(i in Intern, k in Week)
76   sum(j in Rotation)x[i][j][k] <= 1;
77
78 ///Intern Rotation Completiion Constraint (everyone must do one of the rotations)
79 forall(i in Intern)
80   sum(k in Week)x[i][1][k] >= 8;
81
82 forall(i in Intern)
83   sum(k in Week)x[i][2][k] >= 4;
84
85 forall(i in Intern)
86   sum(k in Week)x[i][3][k] == 4;
87
88 forall(i in Intern)
89   sum(k in Week)x[i][4][k] >= 2;
90
91 forall(i in Intern)
92   sum(k in Week)x[i][17][k] >= 2;
93
94 forall(i in Intern)
95   sum(k in Week)x[i][5][k] >= 2;
96
97 forall(i in Intern)
98   sum(k in Week)x[i][6][k] >= 3;
99
100 forall(i in Intern)
101   sum(k in Week)x[i][7][k] >= 3;
102
103 forall(i in Intern)
104   sum(k in Week)x[i][8][k] >= 2;
105
106 forall(i in Intern)
107   sum(k in Week)x[i][9][k] >= 4;
108
109 forall(i in Intern)
110   sum(k in Week)x[i][10][k] >= 3;
111
112 forall(i in Intern)
113   sum(k in Week)x[i][11][k] >= 5;
114
```

```
115 forall(i in Intern)
116   sum(k in Week)x[i][12][k] == 1;
117
118 forall(i in Intern)
119   sum(k in Week)x[i][13][k] == 1;
120
121 //These constraint are the binders for the leave constraints
122
123 forall(i in Intern)
124   sum(k in Week)x[i][14][k] == 1;
125
126 //These for will need to be more flexible than they currently are
127
128 forall(i in 1..6)
129   sum(k in Week)x[i][15][k] ==1;
130
131 forall(i in 1..6)
132   sum(k in Week)x[i][16][k] ==0;
133
134 forall(i in 7..11)
135   sum(k in Week)x[i][15][k] == 0;
136
137 forall(i in 7..11)
138   sum(k in Week)x[i][16][k] == 1;
139
140 ///Intern Rotation Capacity Constraint
141 forall(k in Week)
142   sum(i in Intern)x[i][1][k] <= 2;
143
144 forall(k in Week)
145   sum(i in Intern)x[i][2][k] <= 1;
146
147 forall(k in Week)
148   sum(i in Intern)x[i][3][k] <= 1;
149
150 forall(k in Week)
151   sum(i in Intern)x[i][4][k] <= 1;
152
153 forall(k in Week)
154   sum(i in Intern)x[i][17][k] <= 1;
155
156 forall(k in Week)
157   sum(i in Intern)x[i][5][k] <= 1;
158
159 forall(k in Week)
160   sum(i in Intern)x[i][6][k] <= 1;
161
162 forall(k in Week)
163   sum(i in Intern)x[i][7][k] <= 11;
164
165 forall(k in Week)
166   sum(i in Intern)x[i][8][k] <= 11;
167
168 forall(k in Week)
169   sum(i in Intern)x[i][9][k] <= 2;
170
171 forall(k in Week)
```

```
  sum(i in Intern)x[i][10][k] <= 11;

forall(k in Week)
  sum(i in Intern)x[i][11][k] <= 11;

forall(k in Week)
  sum(i in Intern)x[i][12][k] <= 1;

forall(k in Week)
  sum(i in Intern)x[i][13][k] <= 1;

//Intern Rotation Duration Constraint

forall(i in Intern)
  sum(k in 1..47)y1[i][k] ==1;
forall(i in Intern, k in 1..47)
  8 -(sum(a in 0..7)x[i][1][k + a]) <= M*(1-y1[i][k]);

forall(i in Intern)
  sum(k in 1..51)y2[i][k] ==1;
forall(i in Intern, k in 1..51)
  4 -(sum(a in 0..3)x[i][2][k + a]) <= M*(1-y2[i][k]);

forall(i in Intern)
  sum(k in 1..51)y3[i][k] ==1;
forall(i in Intern, k in 1..51)
  4 -(sum(a in 0..3)x[i][3][k + a]) <= M*(1-y3[i][k]);

forall(i in Intern)
  sum(k in 1..53)y4[i][k] ==1;
forall(i in Intern, k in 1..53)
  2 -(sum(a in 0..1)x[i][4][k + a]) <= M*(1-y4[i][k]);

forall(i in Intern)
  sum(k in 1..53)y4_1[i][k] ==1;
forall(i in Intern, k in 1..53)
  2 -(sum(a in 0..1)x[i][17][k + a]) <= M*(1-y4_1[i][k]);

forall(i in Intern)
  sum(k in 1..53)y5[i][k] ==1;
forall(i in Intern, k in 1..53)
  2 -(sum(a in 0..1)x[i][5][k + a]) <= M*(1-y5[i][k]);

forall(i in Intern)
  sum(k in 1..52)y6[i][k] ==1;
forall(i in Intern, k in 1..52)
  3 -(sum(a in 0..2)x[i][6][k + a]) <= M*(1-y6[i][k]);

forall(i in Intern)
  sum(k in 1..52)y7[i][k] ==1;
forall(i in Intern, k in 1..52)
  3 -(sum(a in 0..2)x[i][7][k + a]) <= M*(1-y7[i][k]);

forall(i in Intern)
  sum(k in 1..53)y8[i][k] ==1;
forall(i in Intern, k in 1..53)
  2 -(sum(a in 0..1)x[i][8][k + a]) <= M*(1-y8[i][k]);
```

```
229
230 forall(i in Intern)
231   sum(k in 1..51)y9[i][k] ==1;
232 forall(i in Intern, k in 1..51)
233   4 -(sum(a in 0..3)x[i][9][k + a]) <= M*(1-y9[i][k]);
234
235 forall(i in Intern)
236   sum(k in 1..52)y10[i][k] ==1;
237 forall(i in Intern, k in 1..52)
238   3 -(sum(a in 0..2)x[i][10][k + a]) <= M*(1-y10[i][k]);
239
240 forall(i in Intern)
241   sum(k in 1..50)y11[i][k] ==1;
242 forall(i in Intern, k in 1..50)
243   5 -(sum(a in 0..4)x[i][11][k + a]) <= M*(1-y11[i][k]);
244
245 //Intern Leave Constraints
246
247 sum(k in 9..50)L1[k] ==1;
248 forall(k in 9..50)
249   11 - (sum(i in Intern)x[i][14][k]) <= M*(1-L1[k]);
250
251 //Avoidance Constraint (1 week rotations during seminar weeks/public holidays)
252
253 forall(i in Intern, j in 12..13)
254   x[i][j][5] ==0;
255
256 forall(i in Intern, j in 12..13)
257   x[i][j][8] ==0;
258
259 forall(i in Intern, j in 12..13)
260   x[i][j][11] ==0;
261
262 forall(i in Intern, j in 12..13)
263   x[i][j][15] ==0;
264
265 forall(i in Intern, j in 12..13)
266   x[i][j][17] ==0;
267
268 forall(i in Intern, j in 12..13)
269   x[i][j][18] ==0;
270
271 forall(i in Intern, j in 12..13)
272   x[i][j][21] ==0;
273
274 forall(i in Intern, j in 12..13)
275   x[i][j][26] ==0;
276
277 forall(i in Intern, j in 12..13)
278   x[i][j][28] ==0;
279
280 forall(i in Intern, j in 12..13)
281   x[i][j][33] ==0;
282
283 forall(i in Intern, j in 12..13)
284   x[i][j][43] ==0;
285
```

```
286 forall(i in Intern, j in 12..13)
287   x[i][j][45] ==0;
288
289 forall(i in Intern, j in 12..13)
290   x[i][j][46] ==0;
291
292 forall(i in Intern, j in 12..13)
293   x[i][j][49] ==0;
294
295
296 }
```