

```

27
28 static ArrayList<Case> liste_des_exclus = new ArrayList<>();
29 static ArrayList<Case> liste_des_trouvés = new ArrayList<>();
30 static ArrayList<Integer> liste_des_lignes = new ArrayList<>();
31 static ArrayList<Integer> liste_des_colonnes = new ArrayList<>();
32 static ArrayList<Integer> liste_des_blocs = new ArrayList<>();
33
34 public static boolean Recherche_des_Interactions() {
35
36     for (int candidat : de_1_à_9) {
37         liste_des_lignes.clear();
38         liste_des_colonnes.clear();
39         liste_des_blocs.clear();
40
41         for (Case kase : liste_des_cases_vides) {
42             if (kase.contient_le(candidat)) {
43
44                 if (!liste_des_colonnes.contains(kase.colonne)) {
45                     if (Interaction_trouvée_dans_la_colonne(kase.colonne, candidat)) {
46
47                         /* Pour l'affichage sur la grille de l'interaction */
48                         Afficher_cette_interaction(candidat);
49                         return true;
50                     }
51                     liste_des_colonnes.add(kase.colonne);
52                 }
53
54                 if (!liste_des_lignes.contains(kase.ligne)) {
55                     if (Interaction_trouvée_dans_la_ligne(kase.ligne, candidat)) {
56
57                         /* Pour l'affichage sur la grille de l'interaction */
58                         Afficher_cette_interaction(candidat);
59                         return true;
60                     }
61                     liste_des_lignes.add(kase.ligne);
62                 }
63
64                 if (!liste_des_blocs.contains(kase.bloc))
65                     if (Interaction_trouvée_dans_le_bloc(kase.bloc, candidat)) {
66
67                         /* Pour l'affichage sur la grille de l'interaction */
68                         Afficher_cette_interaction(candidat);
69                         return true;
70                     }
71                 liste_des_blocs.add(kase.bloc);
72             }
73         }
74     }
75     return false;
76 }
77
78 static boolean Interaction_trouvée_dans_la_colonne(int colonne, int candidat) {
79
80     ArrayList<Integer> liste = new ArrayList<>();
81     Case kase1, kase2, kase3;
82     /* Si il y a une interaction colonne/bloc */
83
84     switch (Compte_des_cases_dans_la_colonne(colonne, candidat, liste)) {
85         /* En retour la "liste" contient des numéros de case */
86
87         case 2:
88             kase1 = la_case(liste.get(0));
89             kase2 = la_case(liste.get(1));
90
91             if (kase1.bloc == kase2.bloc) {
92                 if (Exclus_trouvés_dans_le_bloc(kase1.bloc, candidat, liste_des_cases(liste)))
93                     return true;
94             }
95             break;
96

```

```

97         case 3:
98             kase1 = la_case(liste.get(0));
99             kase2 = la_case(liste.get(1));
100             kase3 = la_case(liste.get(2));
101
102             if (kase1.bloc == kase2.bloc && kase2.bloc == kase3.bloc) {
103                 if (Exclus_trouvés_dans_le_bloc(kase1.bloc, candidat, liste_des_cases(liste)))
104                     return true;
105             }
106         }
107         return false;
108     }
109
110     static boolean Interaction_trouvée_dans_la_ligne(int ligne, int candidat) {
111
112         ArrayList<Integer> liste = new ArrayList<>();
113         Case kase1, kase2, kase3;
114         /* Si il y a une interaction ligne/bloc */
115
116         switch (Compte_des_cases_dans_la_ligne(ligne, candidat, liste)) {
117             /* En retour la "liste" contient des numéros de case */
118
119             case 2:
120                 kase1 = la_case(liste.get(0));
121                 kase2 = la_case(liste.get(1));
122
123                 if (kase1.bloc == kase2.bloc) {
124                     if (Exclus_trouvés_dans_le_bloc(kase1.bloc, candidat, liste_des_cases(liste)))
125                         return true;
126                 }
127                 break;
128
129             case 3:
130                 kase1 = la_case(liste.get(0));
131                 kase2 = la_case(liste.get(1));
132                 kase3 = la_case(liste.get(2));
133
134                 if (kase1.bloc == kase2.bloc && kase2.bloc == kase3.bloc) {
135                     if (Exclus_trouvés_dans_le_bloc(kase1.bloc, candidat, liste_des_cases(liste)))
136                         return true;
137                 }
138             }
139             return false;
140         }
141
142     static boolean Interaction_trouvée_dans_le_bloc(int bloc, int candidat) {
143
144         ArrayList<Integer> liste = new ArrayList<>();
145         Case kase1, kase2, kase3;
146         /* Si il y a une interaction bloc/colonne ou bloc/ligne */
147
148         switch (Compte_des_cases_dans_le_bloc(bloc, candidat, liste)) {
149             /* En retour la "liste" contient des numéros de case */
150
151             case 2:
152                 kase1 = la_case(liste.get(0));
153                 kase2 = la_case(liste.get(1));
154
155                 if (kase1.colonne == kase2.colonne) {
156                     if (Exclus_trouvés_dans_la_colonne(kase1.colonne, candidat, liste_des_cases(liste)))
157                         return true;
158                 }
159                 else if (kase1.ligne == kase2.ligne) {
160                     if (Exclus_trouvés_dans_la_ligne(kase1.ligne, candidat, liste_des_cases(liste)))
161                         return true;
162                 }
163                 break;
164
165             case 3:
166                 kase1 = la_case(liste.get(0));

```

```

167         kase2 = la_case(liste.get(1));
168         kase3 = la_case(liste.get(2));
169
170         if (kase1.colonne == kase2.colonne && kase2.colonne == kase3.colonne) {
171             if (Exclus_trouvés_dans_la_colonne(kase1.colonne, candidat, liste_des_cases(liste)))
172                 return true;
173
174         } else if (kase1.ligne == kase2.ligne && kase2.ligne == kase3.ligne) {
175             if (Exclus_trouvés_dans_la_ligne(kase1.ligne, candidat, liste_des_cases(liste)))
176                 return true;
177         }
178     }
179     return false;
180 }
181
182 static boolean Exclus_trouvés_dans_le_bloc(int bloc, int candidat,
183                                           ArrayList<Case> liste) {
184
185     liste_des_exclus.clear();
186
187     for (int case_de_bloc : de_1_à_9) {
188         Case case_scrutée = la_case(3 * (bloc % 3) + (case_de_bloc - 1) % 3
189                                     + 9 * (3 * (bloc / 3) + (case_de_bloc - 1) / 3));
190
191         if (case_scrutée.contient_le(candidat) && !liste.contains(case_scrutée)) {
192             liste_des_exclus.add(case_scrutée);
193         }
194     }
195
196     if (!liste_des_exclus.isEmpty()) {
197         /* Pour l'affichage de l'interaction */
198         liste_des_trouvés.clear();
199         liste_des_trouvés.addAll(liste);
200         return true;
201     }
202     return false;
203 }
204
205 static boolean Exclus_trouvés_dans_la_colonne(int colonne, int candidat,
206                                               ArrayList<Case> liste) {
207
208     liste_des_exclus.clear();
209
210     for (int ligne : de_1_à_9) {
211         Case case_scrutée = la_case(colonne + 9 * (ligne - 1));
212
213         if (case_scrutée.contient_le(candidat) && !liste.contains(case_scrutée)) {
214             liste_des_exclus.add(case_scrutée);
215         }
216     }
217
218     if (!liste_des_exclus.isEmpty()) {
219         /* Pour l'affichage de l'interaction */
220         liste_des_trouvés.clear();
221         liste_des_trouvés.addAll(liste);
222         return true;
223     }
224     return false;
225 }
226
227 static boolean Exclus_trouvés_dans_la_ligne(int ligne, int candidat,
228                                             ArrayList<Case> liste) {
229
230     liste_des_exclus.clear();
231
232     for (int colonne : de_1_à_9) {
233         Case case_scrutée = la_case(colonne - 1 + 9 * ligne);
234
235         if (case_scrutée.contient_le(candidat) && !liste.contains(case_scrutée)) {

```

```
237         liste_des_exclus.add(case_scrutée);
238     }
239 }
240
241 if (!liste_des_exclus.isEmpty()) {
242     /* Pour l'affichage de l'interaction */
243     liste_des_trouvés.clear();
244     liste_des_trouvés.addAll(liste);
245     return true;
246 }
247 return false;
248 }
249 /*-----*/
250
```