

### EXERCICE 1 : GRAPHIQUE ET GESTION D'ÉVÉNEMENTS

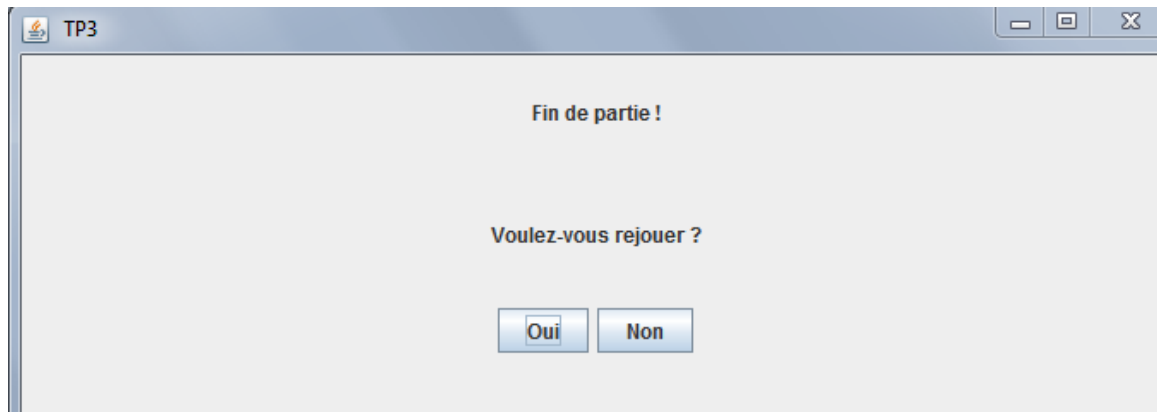
- a. Compléter le fichier *TP2\_Squelette.java* pour créer l'interface graphique ci-dessous. L'interface de la partie droite doit être réalisée avec le gestionnaire de placement **GridBagLayout**.

Avant de coder, faire une grille avec le placement des composants de la partie droite, faire aussi l'arborescence des composants avec les caractéristiques du **GridBagConstraints** – faire valider auprès de l'enseignant(e) ces deux éléments.



Remarque : la classe **Border** permet de mettre des bordures aux Composants (cf. <http://docs.oracle.com/javase/tutorial/uiswing/components/border.html>).

- b. **Les boutons, tout comme les flèches** doivent permettre de gérer les déplacements du disque jaune. Les traces sont persistantes.
- c. Faire évoluer l'IG précédente en ajoutant :
- La gestion du **calcul du score** : en passant sur une case non visitée le score est incrémenté de 10 et en passant sur une case déjà visitée le score est décrémenté de 100.
  - La gestion du **meilleur score** si plusieurs parties se sont déroulées
  - Si l'ensemble du JPanel est parcouru, le dessin s'efface ainsi que le JPanel de droite et l'ensemble est remplacé par le JPanel ci-après



L'appui sur le bouton non, provoque la fermeture de la fenêtre et la sortie de l'application. Si l'utilisateur appuie sur le bouton oui, l'interface initiale réapparaît et une nouvelle partie commence (le meilleur score des anciennes parties doit apparaître).

- iv. Comment faire pour que, si l'utilisateur sort de l'application puis la lance à nouveau, le meilleur score des parties antérieures réalisées soit affiché ?

## EXERCICE 2 :

Reprendre l'exercice 5.b du cours et proposer le code correspondant.

