

Nguyen Philippe

Nourry Charles

Documentation utilisateur : Projet Phine Loops

Notre projet s'utilise en lignes de commandes.

Pour compiler le projet et télécharger tous les composants nécessaires à l'utilisation du projet, il suffit de taper : **mvn install**

Notre projet est divisé en deux parties :

- humbleloop qui contient toutes les classes principales du projet, c'est-à-dire le solveur, le générateur, les pièces...
- app qui contient la partie graphique, utilisant Vaadin, un framework java

Programmes attendus :

Générateur de niveaux :

Afin de générer un puzzle réalisable et mélangé, il suffit de saisir la ligne de commande suivante : `java -jar target/hey-jar-with-dependencies.jar -g wxh --output sortie.txt`

Où w est le nombre de colonnes et h le nombre de lignes du puzzle. sortie.txt correspond au nom du fichier créé, contenant les informations sur la grille que l'on vient de générer.

Le terminal affichera alors : **Running phineloops generator.**

Vérificateur de solution :

Afin de vérifier si une grille est une solution ou non, il suffit de saisir la ligne de commande suivante : `java -jar target/hey-jar-with-dependencies.jar -c file`

Où file est le nom du fichier à tester.

Le terminal affichera alors : **Running phineloops checker.** suivi de **SOLVED : true** si le fichier contient une solution, sinon il affichera **SOLVED : false**

Solveur de niveau :

Afin de vérifier si une grille peut être résolue ou non, il suffit de saisir la ligne de commande suivante : `java -jar target/hey-jar-with-dependencies.jar -s file -output filesolved`

Où file contient les informations sur la grille que l'on souhaite tester et filesolved est le nom du fichier que l'on doit renvoyer avec la solution si elle existe. (filesolved est créé uniquement si la solution existe)

En utilisant le solveur, le terminal affichera plusieurs informations :

Running phineloops solver.

SOLVED : true si le solveur a trouvé une solution, **SOLVED : false** sinon

Partie graphique :

Il est possible de voir l'état d'une grille via une interface graphique. Pour cela, nous avons utilisé Vaadin, un framework java permettant de développer des applications web.

Pour utiliser notre interface, il faut se rendre dans le dossier *app*, puis écrire la ligne de commande **mvn jetty:run**.

En quelques secondes, l'interface est disponible sur serveur local. Il faut alors se rendre à l'adresse <http://localhost:8080/>. On a ensuite la possibilité d'uploader des grilles, afin de les visualiser.

Voici un exemple avec une grille de taille 10x10 résolue :

