**Documentation explicative du programme Scala**

M2 IMSD - LI Bohan & WU Zhuolin

Notre programme lit d'abord le fichier text, qui contient le contenu suivant :

5 5

1 2 N

GAGAGAGAA

3 3 E

AADAADADDA

9 9 N

AAAAAAA

**Description de la logique du programme**

Nous mettons les données lues dans une *liste* ***list*** et construisons un *objet* ***Game***. Nous utilisons ensuite une *méthode* ***Start*** du ***Game*** pour gérer la logique de base. Cette méthode initialise d'abord la carte en lançant la *méthode* ***initMap***, puis exécute les instructions du texte par le biais de ***computeOptions***. Ensuite, nous construisons un *objet* ***Mower*** pour exécuter les instructions correspondantes. Puis nous imprimons l'emplacement actuel de la tondeuse à l'aide de la *méthode* ***printResult*** et appelons la *méthode* ***close*** de la source. Enfin, nous fermons la source et terminons ce programme.

**Explication des classes**

**Classe Game**

1. *Méthode* ***Start*** : permet d'initialiser la mise en page de la matrice entière. Elle comprend une boucle while, qui détermine si les données sont correctes, analyse les instructions de données dans le texte et construit *l'objet* ***Mower***.
2. *Méthode* ***printResult*** : permet d'afficher le numéro de la tondeuse dans ***list*** et sa position.
3. *Méthode* ***computeOptions*** : permet d'exécuter des instructions précises. Les paramètres comprennent un *objet* ***Mower*** et une collection de commandes d'***options***. Une boucle permet à la tondeuse d'exécuter des commandes et de les émettre via la *méthode* ***setOption***.
4. *Méthode* ***initMap***, permet d'initialiser la carte et de construire un array bidimensionnel 10\*10, la tondeuse se déplace à l'intérieur de cet array 10\*10. Ainsi, les coordonnées horizontales et verticales initiales et finales de la tondeuse ne dépassent pas 9.

**Classe Mover**

1. *Méthode* ***setOption*** : si D, la position est changée en D ; si G, la position est changée en G ; si A, la méthode move est appelée ; passe W, x est décrémenté de 1, passe N, y est incrémenté de 1; passe E, x est incrémenté de 1, ; passe S, y est décrémenté de 1.
2. *Méthode* ***move*** : déplace les coordonnées dans les quatre directions WNES.
3. *Méthode* ***changeTo*** : ajoute 1 à l'indice courant si l'instruction passée est D, sinon soustrait 1.

**Résultats de l'épreuve**

文本

描述已自动生成

图形用户界面

描述已自动生成