Une image contenant automate

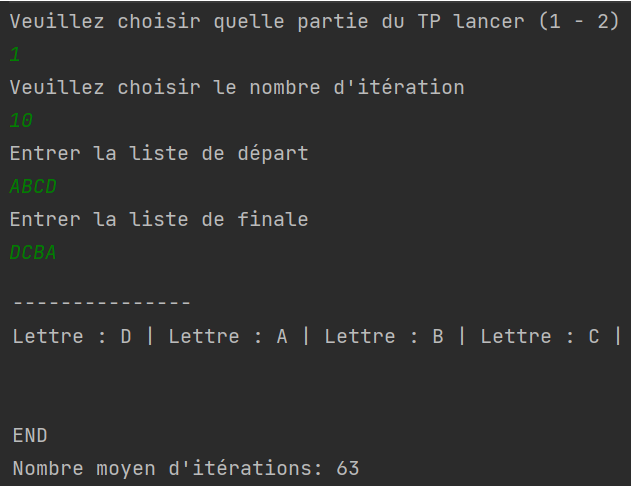
Description générée automatiquement

TP1

BONNET Corentin, MANG Quentin | SMA | 14/11/2021

# Partie 1

Pour lancer la partie 1 du TP, il suffit de lancer la fonction main se trouvant dans la class Main. Ensuite, il faut choisir l’option « 1 » et préciser le nombre d’itération voulu. Ce nombre d’itération va permettre de lancer plusieurs fois le programme avec des listes finales différentes afin de calculer le nombre moyen d’itération.   
Par exemple, si vous entrez une liste finale comme DBAC, il faut entrer une seule itération afin d’avoir cette liste finale à la fin du programme.



L’agent qui doit se déplacer est choisi aléatoirement parmi les 4 agents. En faisant plusieurs tests avec un nombre itérations relativement grand on arrive ainsi à une moyenne de 80 déplacements.

Les classes que nous avons utilisé pour cette première partie sont : *environnement, agent* (classe mère), *agent1* (classe fille) et *main.*

Le fonctionnement de notre programme est simple ; l’agent courant demande à l’agent placé en dessous si c’est bien ce dernier qui soit être sous lui dans la liste finale. Si c’est le cas, alors l’agent courant ne bouge pas. Dans le cas où l’agent en dessous de l’agent courant ne correspond pas à la liste finale, alors ce dernier va, soit pousser l’agent au-dessus de lui, soit se déplacer vers une autre colonne, de manière aléatoire. Ainsi, ces vérifications correspondent à la fonction *isWellPlaced()* dans la classe *Environnement*. Cette fonction prend en paramètre l’agent courant.

Par la suite, ce sont les fonctions *pousser()* et *seDeplacer()* dans la classe *Environnement* qui permettent à l’agent courant de faire une action.

Ces opérations sont répétées jusqu’à obtenir la bonne combinaison correspondante à la liste finale. La fonction permettant de

# Partie 2

Pour lancer la partie 1 du TP, il suffit de lancer la fonction main se trouvant dans la class Main. Ensuite il faut choisir l’option « 1 » et préciser le nombre d’itération voulu pour faire une moyenne sur le nombre d’itération.

Nous avons imaginé plusieurs méthodes permettant d’améliorer le processus utilisé dans la partie 1 du TP1 en optimisant le nombre d’itération moyen. Ces méthodes s’appuient sur une communication locale et simples des agents entre eux. Nous n’avons pas réussi à implémenter correctement les méthodes (les résultats obtenus n’améliorent pas les résultats trouvés dans la partie 1).

La première méthode est une méthode basée sur la notion de priorité. A chaque agent sera attribué une priorité. C’est l’agent avec la plus grande priorité qui va se déplacer. Les priorités sont attribuées par les agents eux-mêmes en communiquant ensemble.

Une deuxième méthode est une méthode qui permet de calculer le meilleur déplacement possible de l’agent. L’agent qui doit se déplacer va « demander aux autres agents dans quelle colonne il doit se déplacer ». C’est-à-dire l’agent va se déplacer dans la colonne où se trouve l’agent supposé être en-dessous.