

L2 – Projet Informatique – Session de rattrapage

Vous développerez en langage C un petit projet de simulation. Vous rendrez le code source de ce projet (dans un fichier .c) ainsi que le compte rendu du développement, la présentation des résultats et leur analyse (dans un document PDF). L'envoi de votre travail se fera par mail à david.hill@uca.fr et je ferai un accusé réception.

Sujet du mémoire de rattrapage : on considère un espace torique en 2 dimensions (grille 2D) de taille 10 x 10 (que vous augmenterez quand le programme sera au point pour étudier les résultats).

Lors de l'initialisation, cette grille 2D est remplie avec une ressource que l'on considère comme du gazon. Le programme doit simuler la tonte de cet espace par pas de temps 't'. Deux possibilités de tonte existent et seront à coder avec 2 tondeuses de type différents placées sur la même grille et démarrant en même temps.

Tondeuse classique : à 't = 0', une tondeuse classique est placée en haut à gauche de la grille et travaillera simplement (la tondeuse avancera en ligne) en se déplaçant à droite vers la cellule suivante sur la même ligne à chaque pas de temps d'une case à la fois. Lorsqu'une ligne est tondue, on passe à la suivante qui se trouve en dessous, mais en partant du point où l'on se trouve (on tond dans un sens, puis dans l'autre).

Robot tondeuse : à 't = 0', un robot tondeuse est placé est placée au milieu de la grille. Il va tondre une cellule à chaque pas de temps et se déplace avec un mouvement aléatoire à chaque pas de temps selon un voisinage de Moore d'ordre 1 (Voir le concept de marche aléatoire ou de mouvement brownien sur Wikipedia si vous êtes intéressé : https://en.wikipedia.org/wiki/Brownian_motion).

Le logiciel fournira une sortie textuelle pour simplifier le développement.

La case où se trouve la tondeuse est marquée par un 'T' – si la tonte est effectuée par le robot tondeur, la case sera marquée par un 'R'. Une case avec du gazon affiche un '~' et quand elle est tondue c'est un '_'.

La simulation s'achève lorsque toutes les ressources sont utilisées (plus rien à tondre).

Faire des observations et des comparaisons entre les 2 types de tondeuses (par exemple combien de cases ont été broutées par chaque type de tondeuse au bout de 'n' itérations ; combien d'itérations faut-il pour tout tondre (avec une tondeuse classique – avec un robot tondeuse aléatoire. Au besoin faire des expériences indépendantes pour consolider vos résultats (score de tonte) avec 95% de confiance. Vous pouvez concevoir différents scénarios : placement des tondeuses à d'autres endroits).

Suivre un style de codage professionnel tel que celui proposé en cours, de même pour la rédaction du rapport, suivez les exemples de qualité qui vous ont été montrés.

Exemple d'affichage de la tonte en mode texte :

t=0	T~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~R~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~	t=1	_T~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~_~~~~~ ~~~~~R~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~	t=2	__T~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~_~~~~~ ~~~~~_R~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~	t=3	___T~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~_~~~~~ ~~~~~R_~~~~~ ~~~~~ ~~~~~ ~~~~~ ~~~~~
		T = 1		T = 2		T = 3	
		R = 1		R = 2		R = 3	

Dans l'exemple ci-dessus, le robot n'est pas très doué, il revient en arrière dans sa marche aléatoire à t=3, du coup à t=4, il n'augmentera pas son score de tonte, car la case sur laquelle il se trouve a déjà été tondue (et par lui-même...). Conseil : dans un 1^{er} temps vous pouvez coder et étudier séparément chaque tondeuse.