

TD 1 : Écriture d'algorithmes

1 Notions à acquérir

Environnement de travail, primitives.

Structures de contrôle : séquence, alternative, itérative.

Composant : spécification d'un composant, définition d'un composant, utilisation d'un composant, arguments d'un composant, type d'un argument

Procédure, fonction, valeur retournée par une fonction, type du résultat d'une fonction

Instruction d'appel de procédure ; appel de fonction.

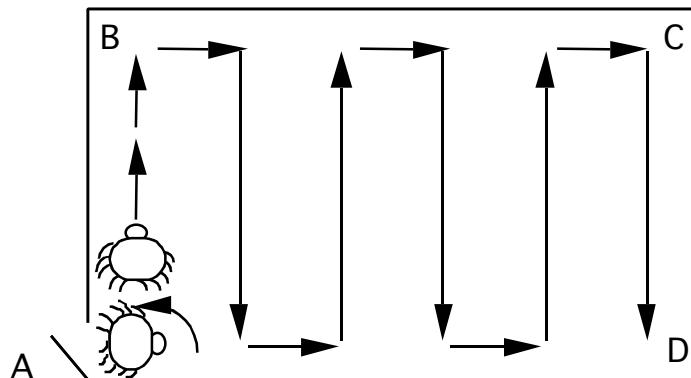
Appel récursif, condition d'arrêt.

2 Un robot nettoyeur

Le robot nettoyeur évolue dans une salle rectangulaire vide ABCD. Il est muni d'un dispositif permettant de tourner sur place, d'avancer et de nettoyer en avançant. Il possède une brosse circulaire nettoyeuse de 50 cm de diamètre et un détecteur d'obstacle.

Pour écrire des algorithmes pour le robot nettoyeur, on doit préciser le domaine algorithmique. Celui-ci est constitué de l'environnement de travail et des primitives que le robot sait exécuter.

L'environnement de travail est formé de la salle et du robot que l'on repère par sa position dans la salle et son orientation. Les états susceptibles d'être modifiés dans cet environnement sont l'état de propreté de la salle, la position du robot et son orientation (sa direction de déplacement).



Les primitives que le robot sait exécuter sont de deux types : des procédures et des fonctions.

2.1 Procédures primitives

Les procédures ont pour effet de modifier l'environnement de travail. Par exemple, lorsque l'ordre TournerQuartDroite est donné au robot, celui-ci va tourner sur lui-même de 90° sur sa droite ; l'orientation du robot se trouvera donc modifiée.

Les procédures que sait exécuter le robot sont :

Procédure **Avancer**

environnement modifié : position du robot

description: le robot avance de 50 cm dans le sens où il est orienté ; dans le cas où il y a un obstacle situé à moins de 50 cm, le robot avance jusqu'à l'obstacle.

Procédure **AvancerNettoyer**

environnement modifié : état de propreté de la salle et position du robot

description : le robot nettoie et avance de 50 cm dans le sens où il est orienté ; dans le cas où il y a un obstacle situé à moins de 50 cm le robot avance et nettoie jusqu'à l'obstacle.

Procédure TournerQuartDroite

environnement modifié : orientation du robot

description : le robot tourne sur lui-même de 90° sur sa droite.

Procédure TournerQuartGauche

environnement modifié : orientation du robot

description : le robot tourne sur lui-même de 90° sur sa gauche.

2.2 Fonctions primitives

Les fonctions ont pour effet de retourner une information sur l'état de l'environnement. Par exemple quand l'algorithme contient l'expression EstMurDevant, l'exécutant de l'algorithme, le robot, va utiliser ses capteurs et va retourner l'information VRAI s'il détecte un mur ou un obstacle juste devant lui dans le sens de sa marche et FAUX sinon.

Fonction EstMurDevant

valeur retournée: de type Booléen

description : retourne l'information VRAI si le robot détecte un obstacle juste devant lui dans le sens de sa marche et FAUX sinon.

Le robot sait aussi exécuter des fonctions similaires à la précédente (EstMurDerrière, EstMurADroite et EstMurAGauche qui retournent VRAI si le robot a un mur derrière lui, à sa droite ou à sa gauche) que l'on utilisera par la suite.

2.3 Écriture de nouvelles procédures

Tenant compte de l'environnement et des possibilités du robot, il est demandé de programmer le robot pour le nettoyage, c'est-à-dire d'écrire l'algorithme permettant au robot de nettoyer la salle. Le robot se situe initialement à la place A et est orienté vers D comme sur la figure. On désire faire nettoyer la salle par bandes en faisant faire au robot des aller-retours parallèles au mur AB jusqu'à arriver en D.

Pour définir une nouvelle procédure, on utilise un langage de description des algorithmes. Ce langage est défini par les structures de contrôle présentées en cours, par les procédures et fonctions primitives du robot, et par les opérateurs booléens OU, ET, NON.

Avant de définir une nouvelle procédure, on doit donner sa spécification : d'éventuelles conditions d'utilisation, en quoi l'environnement est modifié, la description de la procédure en français, et éventuellement ce qu'on sait sur l'environnement après exécution de la procédure (ici la position et l'orientation du robot).

2.3.1 En supposant que le robot est déjà orienté vers le mur BC, on veut écrire tout d'abord la procédure **NettoyerUneBande** qui permet au robot de nettoyer une bande de 50 cm de large par exemple de A vers B. Donner la spécification de NettoyerUneBande, et donner la définition de NettoyerUneBande en langage de description des algorithmes.

2.3.2 Écrire la procédure **NettoyerAllerRetour** qui permet au robot de nettoyer deux bandes contiguës, de 50 cm de large, l'une à partir de sa position initiale jusqu'au mur devant lui et l'autre au retour, de mur à mur, parallèlement à l'orientation initiale du robot.

2.3.3 Écrire la procédure **NettoyerSalleDepuisCoin** qui va permettre au robot de nettoyer la salle à partir de la place A, le robot étant initialement orienté vers D.

La procédure NettoyerSalleDepuisCoin permet de nettoyer toute la salle uniquement si la condition d'utilisation décrite dans sa spécification est bien vérifiée. Que se passe-t-il par exemple si la position initiale du robot est en place A mais orienté vers le mur BC et non vers le mur CD ?

2.4 Écriture d'une nouvelle fonction

En combinant de manière séquentielle les primitives que sait exécuter le robot et en utilisant la structure de contrôle TantQue, on a pu enrichir les possibilités du robot par trois nouvelles procédures NettoyerUneBande, NettoyerAllerRetour et NettoyerSalleDepuisCoin.

On va écrire maintenant de nouvelles fonctions. On rappelle que les fonctions ont pour effet de retourner une information sur l'état de l'environnement.

La seule difficulté de l'écriture d'une fonction par rapport à une procédure est qu'il faut indiquer, à l'intérieur de la fonction, la valeur qu'il faut transmettre comme résultat. Cela se fait par l'instruction **Retourne** qui donne la valeur retournée par la fonction.

La spécification d'une fonction doit préciser le type de la valeur retournée et la description en français de la fonction.

- Écrire une fonction booléenne, EstPasSurMur, qui réponde VRAI si le robot ne touche aucun mur et FAUX sinon.

3 Un robot dessinateur

On considère un robot fixé sur un tableau en un point O. Ce robot sait exécuter deux primitives :

Procédure **TracerCercle**

argument : r, de type entier

environnement modifié : le tableau

description : le robot trace un cercle de centre O et de rayon r.

Procédure **EffacerCercle**

argument : r, de type entier

environnement modifié : le tableau

description : le robot efface sur un cercle de centre O et de rayon r.

3.1 Écriture d'une nouvelle procédure

Écrire une procédure récursive Dessiner(n), où n est un entier strictement positif, qui permet au robot de tracer n cercles concentriques de centre O et de rayons 1, 2, ..., n.

3.2 Comprendre un algorithme

On définit la procédure Mystère(n). Simuler l'exécution de Mystère(3) et donner la spécification de la procédure Mystère(n).

Procédure **Mystère(n)**

Début

Si n = 1 Alors

 TracerCercle(1)

 EffacerCercle(1)

Sinon

 TracerCercle(n)

 Mystère(n-1)

 EffacerCercle(n)

FinSi

Fin