

PRATIQUE DE LA PROGRAMMATION

TD N°2:

Objectif du TD:

Jusqu'ici, vous avez appris à lire un code et à comprendre comment se déroule un programme. Vous savez également comment concevoir des fonctions simples et les appeler.

Pour concevoir un programme, il va vous falloir une étape de plus : La modélisation du problème à résoudre. L'objectif de ce TD est de vous montrer une méthode de modélisation d'un problème : le raffinement.

Elle consiste à modéliser le problème en plusieurs étapes. À chaque étape, on décrit le problème sous forme de sous-problèmes. En passant d'une étape à la suivante, on redécoupe chaque sous-problème. On définit ainsi notre problème de façon de plus en plus précise. On ne s'arrête que quand chaque sous-problème est suffisamment clairement décrit pour que l'on puisse concevoir un code qui prenne en charge ce sous-problème.

ENONCE

Dans la première partie de ce TD, nous allons voir comment nous avons pu obtenir le code du programme du lièvre et de la tortue que vous connaissez. Dans la seconde partie de ce TD, vous allez raffiner un nouveau problème jusqu'à pouvoir le programmer !

PARTIE 1 : Modélisation du problème de la course du lièvre et de la tortue.

Niveau 0 : simulation de la course du lièvre et de la tortue par un ordinateur

Niveau 1

1. Départ de la course
2. Faire avancer la tortue sur le plateau en fonction des règles établies
3. Faire avancer le lièvre sur le plateau en fonction des règles établies
4. Fin de la course

Niveau 2

1. Initialiser le jeu
2. Faire avancer la tortue sur le plateau en fonction des règles établies
3. Faire avancer le lièvre sur le plateau en fonction des règles établies
4. Si le lièvre ou la tortue arrive en fin de course, la partie s'arrête
5. Sinon On recommence à l'étape 2

Niveau 3

1. Initialiser le jeu : course de 70 cases, lièvre et tortue au point de départ
2. Tant que (la course n'est pas terminée)
 - a. Modifier la position de la tortue
 - b. Modifier la position du lièvre
 - c. Afficher les joueurs à leur nouvelle position

Niveau 4

Course Terminée : si la position du lièvre ou la position de la tortue est au-delà de la case 70.

Déplacement de la tortue

1. On tire une valeur aléatoire entre 1 et 10
2. Si la valeur est inférieure ou égale à 5, la tortue avance de 3 cases (un pas rapide)
3. Si la valeur est comprise entre 6 et 7, la tortue avance de 6 cases (glissade)
4. Si la valeur est supérieure ou égale à 8, la tortue avance de 1 case (un pas lent)

Déplacement du lièvre

1. On tire une valeur aléatoire entre 1 et 10
2. Si la valeur est inférieure ou égale à 2, le lièvre dort et sa position ne change pas (dort)
3. Si la valeur est comprise entre 3 et 4, le lièvre avance de 9 cases (grand saut)
4. Si la valeur est comprise entre 5 et 7, le lièvre avance de 1 case (petit saut)
5. Si la valeur est égale à 8, le lièvre recule de 9 cases (grande glissade)
6. Si la valeur est égale à 10, le lièvre recule de 2 cases (petite glissade)

Affichage de la course

1. Afficher le plateau de la course
2. Afficher la position de la tortue
3. Afficher la position du lièvre

PARTIE 2 : Modélisation d'un nouveau problème

Sur le même principe, on souhaite modéliser le jeu de MasterMind. En fait, il s'agit d'un MasterMind simplifié : 10 tours maximums pour deviner une combinaison de 5 pions de couleurs (4 couleurs possibles). Le joueur ayant défini la combinaison à trouver indique à chaque tour à son adversaire combien de pions de la bonne couleur sont bien placés.

1. Avec l'approche par raffinages successifs, détaillez au maximum les étapes nécessaires pour programmer un jeu de MasterMind
2. Donnez le programme C équivalent.