

TD N° 8 d'Algorithmique

Exercice 1 : Bataille Navale

Le jeu de la bataille navale est un jeu à 2 joueurs où chaque joueur place de façon cachée des navires sur une grille de 10x10 (A-J en horizontal, 1 à 10 en vertical).

Chaque joueur dispose de 1 porte-avions (occupant 5 cases), 1 croiseur (4 cases), 2 contre-torpilleurs (3 cases), 1 torpilleur (2 cases).

	(x) 0	1	2	3	4	5	6	7	8	9
(y) 0	X									
1	X		X	X	X					
2	X									
3	X						X	X		
4	X									
5									X	
6									X	
7		X	X	X					X	
8									X	
9										

A tour de rôle, un joueur annonce la case visée, s'il touche un bâtiment ennemi, il peut rejouer, sinon c'est au tour de l'adversaire.

Le 1^{er} joueur à avoir éliminé les bateaux de l'adversaire gagne.

Question 1. Structure de donnée

Les navires n'occupent que 17 cases sur les 100 possibles.. Utiliser une grille serait une perte de place mémoire. On décide de se passer de grille.

Question 1. Proposez des types de données structurés :

- **TypeNavire** pour le type de navire (contenant nom (texte), taille (entier) et nombre de pions possibles (nb entier))

- **Navire** pour ... un navire (contenant son type, sa position, son orientation (1 pour Nord, 2 pour Est, 3 pour Sud et 4 pour Ouest)).

On définit par la suite les types de tableaux de TypeNavire et de Navire :

```
type TabType = tableau de TypeNavire  
type TabNavire = tableau de Navire
```

Créez la fonction **creerTypeNavire(nom:Texte, taille:entier, nb:entier):TypeNavire** qui crée un type de navire et lui affecte nom, taille et nb d'éléments.

On définit ensuite une constante **pieces** qui contient les types de pièces de jeu possibles :

```
const pieces:TabType = nouveau tableau de TypeNavire de dimension 4  
pieces[0] = creerTypeNavire("porte avion", 5, 1)  
pieces[1] = creerTypeNavire("croiseur", 4, 1)  
pieces[2] = creerTypeNavire("contre torpilleur", 3, 2)  
pieces[3] = creerTypeNavire(" torpilleur", 2, 1)
```

Question 2. Placement des navires

Créez la fonction `placerNavires(tab : TabNavire)` qui, à l'aide du tableau des pièces, demande le placement de chaque navire (x,y et orientation) à l'utilisateur et les stocke dans tab (que l'on supposera assez grand pour stocker les pièces)

Question 3. Vérifier si touche

Créez la fonction `estTouche(x:entier, y:entier, n:Navire):booléen` qui retourne si le navire n est touché ou non par un tir en x,y.

Par exemple si n est le porte-avion de la grille précédente, il possède comme attribut `type=pieces[0]`, `x=0`, `y=0`, `orientation=3` (le sud). (Il est possible de connaître sa taille : `type.taille`)

Un tir en 1,0 le manque. Un tir en 0,3 le touche..

Autre exemple, le contre-torpilleur posé en (7,3) de direction 4 (Ouest) : un tir en 8,3 le manque, un tir en 6,3 le touche.

Créez la fonction `navireTouche(x:entier, y:entier, tab:TabNavire):Navire` qui retourne le navire de tab touché par un tir en x,y (retourne vide sinon).

Question 4. Evaluer les touches

Il manque des champs à la structure navire pour connaître son état.. Ajoutez les champs : `entier nbTouches`, `booléen coule` et `touches` un tableau d'entiers..

Lors du placement d'un navire il faudra ajouter une ligne pour créer ce tableau touches qui doit posséder autant de cases que la taille du navire.

Ce tableau contient les touches sous le format (100.x + y)

Créez la fonction `tir(x:entier, y:entier, tab:TabNavire)` qui :

- vérifie si un navire a été touché, et dans l'affirmative :
 - Incrémente le nombre de touches
 - Stocke cette touche,
 - Vérifie s'il n'a pas été coulé.

Question 5. Poser automatiquement des navires

Créez la fonction `creerNavire(type:TypeNavire, x:entier, y:entier, orientation:entier):Navire` qui crée un navire et lui affecte type, coordonnées et orientation et le retourne.

Créez la fonction `afficherNavire(n:Navire)` qui affiche type, coordonnées et orientation, état, nb de touches (et les affiche le cas échéant).

Créez la fonction `autoplacerNavires(tab : TabNavire)` qui, à l'aide du tableau des pièces, place automatiquement les navires : sans qu'ils sortent de la grille 10x10, et sans qu'ils se croisent.

Question 6. Le jeu

Créez la fonction `jeu()` qui demande le positionnement des navires de l'utilisateur, place automatiquement des navires et lance le jeu entre l'humain et la machine jusqu'à ce qu'il y ait un gagnant.