Institut National des Télécommunications

Contrôle des Connaissances 2

Introduction à la Programmation	**Module : INF3002**
Date : 14/05/2008	**Durée : 1h30**
Coordonnatrices : **S. Chabridon et C. Lecocq**	**Documents autorisés : ceux distribués en cours et notes personnelles**

Avertissements

1/ Lisez attentivement le sujet.

2/ Soyez clairs et précis dans vos réponses.

3/ Soignez la présentation.

3/ Respectez les noms de variables et de paramètres demandés

4/ Traitez les questions dans l'ordre

5/ Barème indicatif

Question 1 : 2 points.

Question 2 : 5.5 points.

Question 3 : 3.5 points.

Question 4 : 4 points.

Question 5 : 3 points.

Question 6 : 2 points.

Bataille navale - Il vous est demandé de programmer une version simplifiée d'un jeu de bataille navale entre deux joueurs, jouant à tour de rôle sur la même machine (et ne regardant pas l'écran lorsque l'autre joueur place ses vaisseaux !!).

Le programme commence par créer deux grilles (tableau de 100 cases, soit 10 lignes et 10 colonnes) contenant des nombres entiers et chaque joueur devra indiquer où il place ses vaisseaux. Ensuite, les joueurs essaient à tour de rôle de couler les vaisseaux adverses.

Deux types de vaisseaux peuvent être placés : bateau et sous-marin. Un bateau occupe 1 seule case et un sous-marin 2 cases. Pour différencier les vaisseaux, un bateau est indiqué par le chiffre 1 et un sous-marin par le chiffre 2. Chaque joueur doit placer 5 vaisseaux au total, 3 bateaux et 2 sous-marins.

Exemple de grilles :

Grille du joueur 1

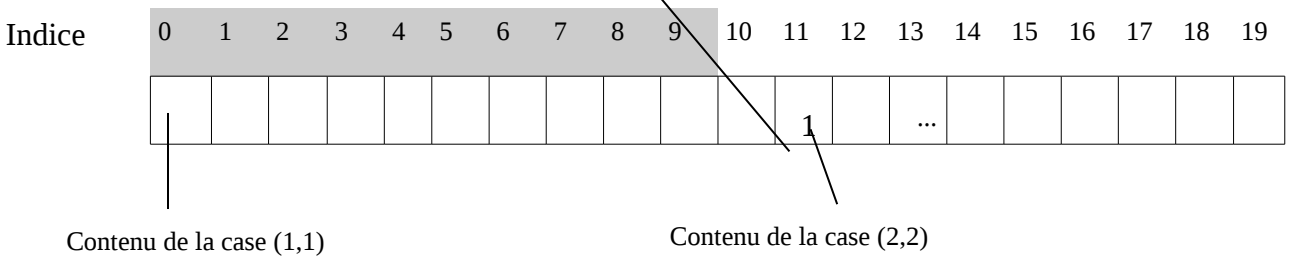
	1	2	3	4	5	6	7	8	9	10
1										
2		1								
3										
4					1					
5										
6										
7							2	2		
8		2	2							
9										
10									1	

Grille du joueur 2

	1	2	3	4	5	6	7	8	9	10
1	1									
2										
3										
4										1
5				2	2					
6										
7								1		
8										
9	2	2								
10										

Les grilles ne sont en fait pas stockées sous forme de tableau à deux dimensions. Les lignes d'une grille sont mises bout à bout dans un tableau à une dimension (tableau de 100 cases).

Indication pour la numérotation des cases des 2 premières lignes de la grille du joueur 1 :



Les cases d'indice 0 à 9 correspondent à la ligne 1 de la grille. Les cases d'indice 10 à 19 correspondent à la ligne 2 de la grille etc.

Le contenu de la case (i, j) est placé à l'indice **(i - 1) * 10 + j - 1** dans le tableau.

La grille contient la valeur 1 dans la (2,2).

Cette valeur est donc placée dans la case d'indice 11 du tableau.

Questions

Question 1 : définition des structures de données

Définir 2 tableaux de 100 entiers nommés `grille1` et `grille2.` Pour cela vous devez écrire sur une feuille séparée une méthode « main() ». Vous écrirez la création des tableaux dans cette méthode.

Pour chacune des questions suivantes, écrire la méthode demandée sur votre copie puis compléter le programme principal (main) permettant d'appeler cette méthode sur la feuille séparée. Conseil : Essayer d'écrire le main même si les réponses aux autres questions ne sont pas complètes.

Question 2 : Remplir les grilles

Écrire une méthode permettant de saisir (lire) les vaisseaux d'un joueur. L'utilisateur doit placer tout d'abord les 3 bateaux, un par un, puis les 2 sous-marins. Pour chaque vaisseau, demander à l'utilisateur la ligne et la colonne où il souhaite le placer. S'il s'agit d'un bateau, mettre le nombre 1 dans la case correspondante du tableau t. Pour un sous-marin, mettre le nombre 2 dans la case indiquée et aussi dans la case placée juste à droite (en ajoutant 1 à l'indice).

Indication : Utiliser la formule indiquée dans l'exemple précédent pour calculer l'indice de la case à remplir. La case (i,j) de la grille correspond à la case d'indice **(i - 1) * 10 + j - 1** dans le tableau.

Hypothèse : On supposera que l'utilisateur ne fait pas d'erreur de saisie et entre toujours des valeurs correctes.

2.1 Écrire la méthode permettant de saisir une grille. Le prototype (l'entête) de la méthode est le suivant :

```
static void saisirGrille(int [] t){
```

2.2 Compléter le programme principal (main) afin de faire 2 appels à la méthode que vous venez de définir pour saisir les tableaux `grille1 et grille2` définis à la question 1. Afficher avant chaque appel un message pour indiquer quel joueur doit saisir sa grille.

Question 3 : Afficher une grille

Écrire une méthode pour afficher le contenu d'une grille en lignes et en colonnes en passant à la ligne suivante à la fin de chaque ligne. Prévoir 2 indices pour parcourir le tableau contenant la grille et utiliser la formule donnée précédemment.

3.1 Écrire une méthode permettant d'afficher le contenu du tableau donné en paramètre sous la forme d'une grille. Le prototype (l'entête) de la méthode est le suivant :

```
static void afficherGrille(int[] t)
```

3.2 Compléter le programme principal (main) afin de faire 2 appels à la méthode que vous venez de définir pour afficher les tableaux `grille1 et grille2`. Afficher avant chaque appel un message pour indiquer quelle grille va être affichée.

Question 4 : Valeur totale d'une grille

À chaque fois qu'un joueur touche un vaisseau adverse, la case correspondante de la grille est mise à 0. Ainsi, pour savoir si un joueur a gagné, il suffit de vérifier que la somme de toutes les valeurs de la grille adverse est nulle. Cette méthode permet de faire ce calcul pour une grille donnée.

4.1 Écrire la méthode permettant de calculer la valeur totale d'une grille passée en paramètre. Le prototype (l'entête) de la méthode est le suivant :

```
static int calculerTotal(int[] t)
```

NB : Cette méthode n'est pas appelée directement dans le `main` mais sera appelée à la question 6.

Question 5 : Touché - Coulé

Écrire une méthode pour tester le contenu d'une case dont le numéro de ligne i et le numéro de colonne j sont donnés en paramètres. Commencer par calculer l'indice de la case à tester (utiliser la formule indiquée à la question 2).

Tester le contenu de cette case dans le tableau stockant la grille considérée. Si la case contient 1, un bateau vient d'être coulé : mettre la case à 0 et afficher le message « Coulé ! ».

Si la case contient 2, il s'agit d'un sous-marin placé sur 2 cases. Il faut alors tester les cases situées à droite et à gauche (ajouter ou enlever 1 à l'indice de la case). Si aucune des 2 cases voisines ne contient la valeur 2, alors le sous-marin vient d'être coulé et il faut afficher le message « Coulé ! ». Par contre, si l'une des cases voisines contient un 2, il faut afficher simplement le message « Touché ! ».

5.1 Écrire la méthode permettant de tester le contenu de la case (i, j) et afficher les messages demandés. Le prototype (l'entête) de la méthode est le suivant :

```
static void testerUneCase(int[] t, int i, int j)
```

NB : Cette méthode n'est pas appelée directement dans le `main` mais sera appelée à la question 6.

Question 6 : Chacun son tour

Cette méthode gère la logique de jeu. Elle reçoit en paramètres les 2 grilles créées à la question 1 et demande à chaque joueur à tour de rôle de choisir une case dans la grille adverse. Le déroulement de la méthode est le suivant :

1. Afficher un message demandant au joueur 1 de donner la ligne et la colonne choisies.
2. Appeler la méthode `testerUneCase(...)` définie à la question 5 en passant en paramètres la grille du joueur 2 et la ligne et la colonne choisies.
3. Vérifier ensuite si le joueur a gagné en appelant la méthode `calculerTotal(...)` définie à la question 4 et en testant si le résultat obtenu donné par cette méthode est nul. Dans ce cas, afficher un message indiquant que le joueur 1 a gagné et que la partie est terminée.
4. Si la valeur totale de la grille n'est pas 0, c'est au tour du joueur 2 qui peut à son tour choisir une case dans la grille du joueur 1 en suivant les mêmes étapes.

La méthode doit ensuite indiquer si la partie est terminée ou non à l'aide du résultat qu'elle renvoie.

6.1 Écrire la méthode permettant de jouer un tour. Le prototype (l'entête) de la méthode est le suivant:

```
static boolean jouerUnTour(int[] g1, int[] g2)
```

6.2 Compléter le programme principal (main) pour appeler la méthode que vous venez de définir **tant que** la partie n'est pas terminée.