

Matière

- La boucle *for*

Objectifs

- Savoir utiliser la boucle *for* dans un programme simple
- Savoir résoudre les problèmes classiques que sont les calculs de somme, de moyenne, de minimum et de maximum d'une suite de nombres
- Comprendre l'impact d'un *for* dans un *for*

Exercices obligatoires

A Le « for »

A1.a Ecrivez la classe *SommeCinqEntiers*. Cette classe lit 5 entiers au clavier et affiche leur somme.

A1.b. Que faut-il modifier au programme précédent pour traiter 100 données au lieu de 5 ? Ecrivez la classe *SommeCentEntiers*.

A2 Écrivez un programme qui affiche à l'écran la table de multiplication demandée par l'utilisateur.

Ex : si l'utilisateur demande la table de multiplication de 3, le programme affiche :

```
1 x 3 = 3
2 x 3 = 6
3 x 3 = 9
...
10 x 3 = 30
```

Appelez votre classe *TableDeMultiplications*.

B Des « if » dans des « for »

B1 Écrivez un programme qui lit 10 entiers au clavier et ensuite affiche le nombre d'entiers négatifs.

Nommez votre classe *NombreNegatifsParmi10*.

B2 Écrivez un programme qui lit 5 entiers au clavier et ensuite affiche la somme des nombres positifs (≥ 0) et la somme des nombres négatifs.

Nommez votre classe *SommesNegatifsEtPositifs*.

B3 Un étudiant a passé plusieurs examens, cotés chacun sur 20 (à ne pas vérifier). Complétez la classe *Statistiques*. Elle demande le nombre d'examens passés, ensuite lit les cotes et affiche la cote la plus élevée.
On voudrait que cette classe affiche également la cote la plus basse et la moyenne

B4 Aux éliminatoires du championnat du monde de gymnastique, chaque concurrent est coté pour la présentation de son exercice, sur 10 points, par 8 membres du jury.
On élimine de ces cotes, la plus haute et la plus basse. Le résultat d'un concurrent est égal à la moyenne des 6 cotes restantes.
Pour passer en finale, il faut un résultat d'au moins 8/10.
Écrivez un programme qui, pour un concurrent, lit les cotes sur 10 (à ne pas vérifier) du jury, affiche son résultat et signale s'il est sélectionné pour la finale.
Appelez votre classe *Championnat*.

B5 Un étudiant a passé 5 examens, cotés chacun sur 10 (à ne pas vérifier).
Écrivez un programme qui lit ces 5 cotes et affiche ses deux meilleures cotes
Testez **bien** cette classe avec les nombres suivants :

cote1	cote2	cote3	cote4	cote5	Résultats attendus
8	2	9	3	5	9 et 8
8	2	7	3	5	8 et 7
8	4	7	5	8	8 et 8
8	7	6	5	4	8 et 7
4	5	6	7	8	8 et 7

Nommez votre classe *DeuxMeilleuresCotes*.

C Des for dans des for

C1 Code génétique

Le code ADN est composé de 4 nucléotides qui portent les lettres TGCA. Pour des raisons de facilité, nous les numérotions 1, 2, 3 et 4. On désire générer toutes les combinaisons de ces 4 chiffres de longueur 2 :

11
12
13
14
21
22
23
24
31
32
33
34
41
42
43
44

Ecrivez la classe *TGCA* qui affiche toutes ces combinaisons.

C2 Ecrivez la classe *AffichageTablesDeMultiplications* pour qu'elle affiche à l'écran toutes les tables de multiplication de 1 à 10 :

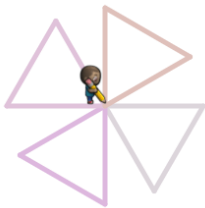
```

1 2 3 4 5 6 7 8 9 10
2 4 6 8 10 12 14 16 18 20
3 6 9 12 15 18 21 24 27 30
4 8 12 16 20 24 28 32 36 40
5 10 15 20 25 30 35 40 45 50
6 12 18 24 30 36 42 48 54 60
7 14 21 28 35 42 49 56 63 70
8 16 24 32 40 48 56 64 72 80
9 18 27 36 45 54 63 72 81 90
10 20 30 40 50 60 70 80 90 100

```

C3 La tortue : le retour !

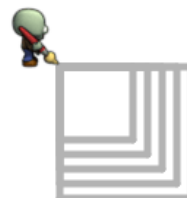
La classe *DessinFleur* vous est donnée en exemple.
Elle a été présentée lors du cours théorique.



C3.1 Ecrivez la classe *DessinVentilateur*.



C3.2 Ecrivez la classe *DessinCarresEmboités*.



C4 « les petits dessins »

On n'utilise pas la tortue !

`System.out.print('X') ;` Pour écrire une croix
`System.out.print(' ') ;` Pour mettre un espace (un « blanc »)
`System.out.println() ;` Pour passer à la ligne

Complétez les classes *Carre*, *TriangleV1*, *TriangleV2*, *TriangleV3*, *Pyramide* et *Losange*.

Dans cet exercice, nous vous demandons de construire 5 algorithmes de dessin (voir figures ci-dessous). Vos algorithmes devront permettre de construire des dessins de taille variable. Les modèles de dessins ci-dessous correspondent tous à une taille de 4. On peut remarquer que la taille de chaque ligne et de chaque colonne est un multiple de 4.

Carré	Triangle version 1	Triangle version 2
XXXX XXXX XXXX XXXX	X XX XXX XXXX	XXXX XXX XX X
Triangle version 3	Pyramide	Losange
XXXX XXX XX X	X XXX XXXXX XXXXXXX	XXXXXXXXX XXX XXX XX XX X X X X XX XX XXX XXX XXXXXXXXX

Exercices supplémentaires

B6. Écrivez un programme qui lit 10 entiers au clavier et ensuite affiche le nombre d'entiers positifs et le nombre d'entiers négatifs rencontrés.

B7 Écrivez un programme qui lit les noms et les résultats (sur 100) des 10 participants à un concours. Le programme affiche le nom du vainqueur. En cas d'ex-æquo, il en affiche un seul, mais signale la présence d'ex-æquo.

B8 Ecrivez un programme qui propose 3 exercices de multiplications de 2 nombres entiers tirés au hasard entre 0 et 10.
Pour chaque multiplication l'utilisateur est invité à donner la réponse. Si la réponse entrée au clavier n'est pas bonne, le programme affiche la bonne réponse, sinon il affiche le message suivant : « Bravo ! »

C3 (suite) Ecrivez la classe *DessinJeuxOlympiques* qui dessine le logo des jeux olympiques.



C4 (suite) Ecrivez les classes *PyramideV2* et *PyramideV3* :

Pyramide version 2	Pyramide version 3
1	7
1 2 1	8 9 8
1 2 3 2 1	9 0 1 0 9
1 2 3 4 3 2 1	0 1 2 3 2 1 0

Pour la pyramide version 3, on lit le chiffre du sommet en plus de la taille.

Exercices défis

C5 Afficher le calendrier d'un mois donné.

Chaque semaine doit être affichée sur une ligne.

Vous devez introduire au clavier :

- la longueur du mois (28, 29, 30 ou 31 jours) ;
- quel jour tombe le 1^{er} du mois concerné (1 si c'est lundi, ...)

Exemple :

Lu	Ma	Me	Je	Ve	Sa	Di
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28				

C2 (amélioré)

Améliorez la classe *AffichageTablesDeMultiplications* pour qu'elle affiche à l'écran toutes les tables de multiplication de 1 à 10 selon le format :

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

C6

Ecrivez un programme qui propose des exercices d'arithmétique.

C'est l'utilisateur qui choisit le nombre d'exercices qu'il souhaite faire.

Le programme mélange (au hasard) des additions, des soustractions, des multiplications et des divisions.

Pour chaque exercice, l'utilisateur est invité à donner la réponse. Si la réponse n'est pas bonne, le programme affiche la bonne réponse.

A la fin, le programme affiche le pourcentage de bonnes réponses.

ATTENTION : L'utilisateur à qui s'adressent ces exercices ne doit manipuler que des entiers compris entre 1 et 100.

Les divisions ne peuvent avoir comme réponse des nombres qui ne sont pas des entiers.

(8/3 = ?)

Il faudrait voir apparaître des multiplications, des additions, de soustractions et des divisions avec des entiers compris entre 1 et 100, mais toujours avec une réponse comprise entre 1 et 100.

Ex : 12 x 6 32 + 54 76 - 15 75/5 ...