



Projet Déménagement

Étape 4 : Rangement des cartons

Modalités

- Travail en îlot et en autonomie
- Production individuelle

Objectifs de l'activité

- Apprendre à travailler sur des tableaux

Sujet

Les déménageurs dispatchent les cartons dans les différents bureaux du nouveau local. Il y a 15 bureaux différents. Les cartons arrivent un par un, et l'on veut au final savoir combien l'on a de cartons dans chaque bureau.

Consignes

- Dans le plan des locaux :
 - Chaque bureau est représenté par une case.
 - Les bureaux (cases) sont numérotés de 0 à 14 (soit 15 bureaux)
 - Le nombre maximum de cartons que peut contenir chaque bureau (case) est indiqué.
- Par îlot, créez un plan à l'échelle, permettant de stocker exactement le bon nombre de cartons dans votre îlot (représentés par des Lego, dominos, ...), avec des couloirs et des portes suffisamment grandes pour que les cartons transitent.
- Simuler à l'aide du plan et des éléments de déménagement disponibles dans votre îlot, chacune des quatre situations proposées.
- Pour chaque situation :
 - Lister les actions réalisées
 - Décrire la démarche utilisée et élaborer le pseudocode
- Appréhender la syntaxe de manipulation de tableaux sous Processing en faisant les tutos d'Openclassroom sur le langage Java sur le chapitre tableaux.

On utilisera des tableaux pour représenter les bureaux. Dans un premier temps, on utilisera un tableau contenant le nombre de cartons stocké dans chaque bureau (chaque case), et un autre tableau contenant pour chaque bureau, sa capacité totale (nombre maxi de cartons).

Dans une seconde version optionnelle, on peut aussi utiliser un tableau de tableau (tableau contenant dans chaque case, un tableau de 2 cases avec le nombre de cartons stockés et le nombre max).
- Pour chacune des situations proposées, coder l'algorithme correspondant

Situations proposées

Situation 1 (notion de boucle)

Remplir le premier bureau au maximum de sa capacité, puis le deuxième bureau, puis le troisième, etc.

Situation 2 (notion de boucle)

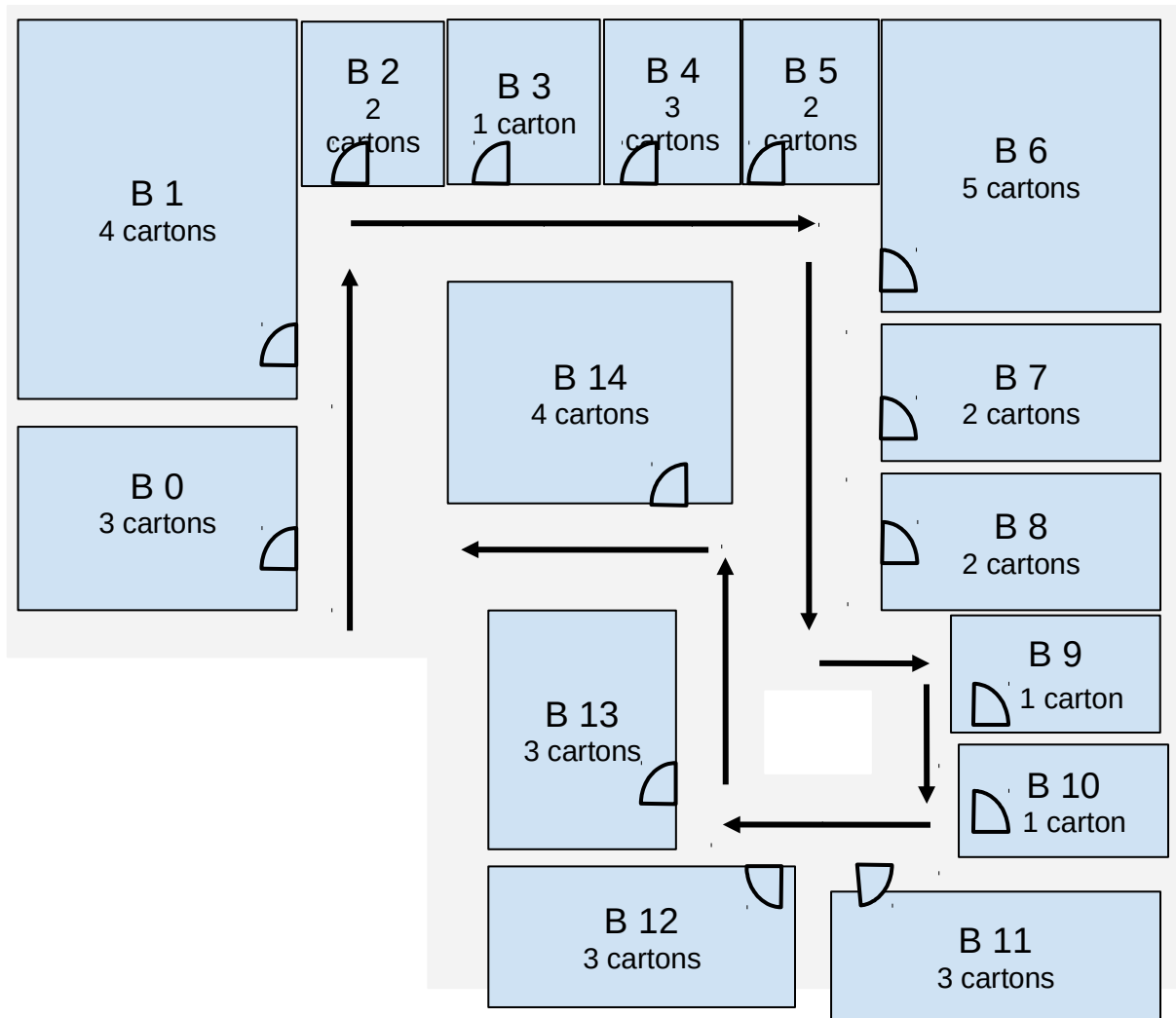
Déposez un carton dans le premier bureau puis un carton dans le deuxième bureau, etc. jusqu'au bureau 14. Recommencez jusqu'à ce que la capacité de chaque bureau soit atteinte

Situation 3 (notion de variable temporaire)

Les déménageurs ont inversé les cartons dans deux bureaux : le 5 et le 10. Procédez à l'échange.

Ressources

- <https://openclassrooms.com/fr/courses/1397856-processing/1398209-fonctions-2-2>
(uniquement partie tableaux)
- <https://processing.org/reference/>
- <https://openclassrooms.com/fr/courses/26832-apprenez-a-programmer-en-java/20998-les-tableaux>
- Plan des bureaux



Production

- Pour chacune des 3 situations proposées, fournir les algorithmes correspondants (en pseudocode), et les programmes commentés correspondants. Les programmes doivent permettre d'afficher à la fin le contenu (nombre de cartons) des différents bureaux.

Pour aller plus loin : Tri des cartons

Modalités

- Travail individuel.
- Travail optionnel, destiné uniquement à ceux qui vont vite...
- Production individuelle

Objectifs de l'activité

- Apprendre à travailler sur des tableaux

Sujet

Les cartons sont pris un par un et pesés (le poids est donné comme précédemment par *float poids = random(8);*). Les poids sont stockés dans un tableau

Consignes

- Nous allons trier ce tableau de poids. Pour cela, nous utiliserons une méthode classique de tri : le tri par remontée des bulles.
- Principe du tri par remontée des bulles : on parcourt le tableau de valeurs en comparant les éléments deux à deux. Si le plus grand est avant le suivant, on inverse les éléments dans le tableau.

On fait autant de parcours du tableau que nécessaire pour que le tableau soit trié.

Exemple : État du tableau pendant le parcours
premier parcours (en sature les éléments comparés 2 à 2)

9		1		1		1
1	>>	9	>>	4	>>	4
4		4		9		2
2		2		2		9

deuxième parcours

1		1		1		1
4	>>	4	>>	2	>>	2
2		2		4		4
9		9		9		9

il y a eu des inversions pendant le deuxième parcours, on recommence un troisième parcours

1		1		1		1
2	>>	2	>>	2	>>	2
4		4		4		4
9		9		9		9

Il n'y a pas eu d'inversions pendant le troisième parcours, le tableau est donc trié.

- Écrire sous forme de pseudocode l'algorithme de tri de ce tableau.

- Coder avec Processing le programme qui remplit le tableau des poids, l'affiche avant le tri, effectue le tri, puis l'affiche après le tri.

Ressources

- <https://openclassrooms.com/fr/courses/1397856-processing/1398209-fonctions-2-2>
(uniquement partie tableaux)
- <https://processing.org/reference/>
- <https://openclassrooms.com/fr/courses/26832-apprenez-a-programmer-en-java/20998-les-tableaux>
- <https://www.toptal.com/developers/sorting-algorithms/bubble-sort>
- http://lwh.free.fr/pages/algo/tri/tri_bulle.html
- <https://www.youtube.com/watch?v=lyZQPjUT5B4>

Production

- Algorithme du tri en pseudocode
- Code Processing (.pde) sur votre dossier personnel.