

Configuration de Classes pour GLIA-AIRLINES

La compagnie aérienne GLIA-AIRLINES souhaite équiper sa flotte d'avions avec des séparateurs de classes, permettant ainsi des configurations et des tailles de classes différentes.

La compagnie propose des sièges dans les catégories "First Class", "Business Class" et "Economy Class". De plus, elle offre la location de ses sièges à d'autres compagnies aériennes.

Cependant, les séparateurs sont coûteux et occupent un espace non négligeable à bord de l'avion. L'objectif est de positionner n séparateurs dans un avion ayant une capacité de m blocs de sièges. Il est impératif de ne pas installer de séparateurs au niveau des sorties de secours (au moins une sortie). De plus, il est essentiel que la classe à l'avant de l'avion dispose d'au moins deux blocs.

Afin de permettre des configurations avec des tailles de classes variées, il est nécessaire de garantir que les distances entre les séparateurs sont différentes.

La figure 1 illustre le problème avec un avion d'une capacité de $m = 11$ blocs, $n = 5$ séparateurs à installer, et une seule sortie de secours à la position $exit_1 = 3$.

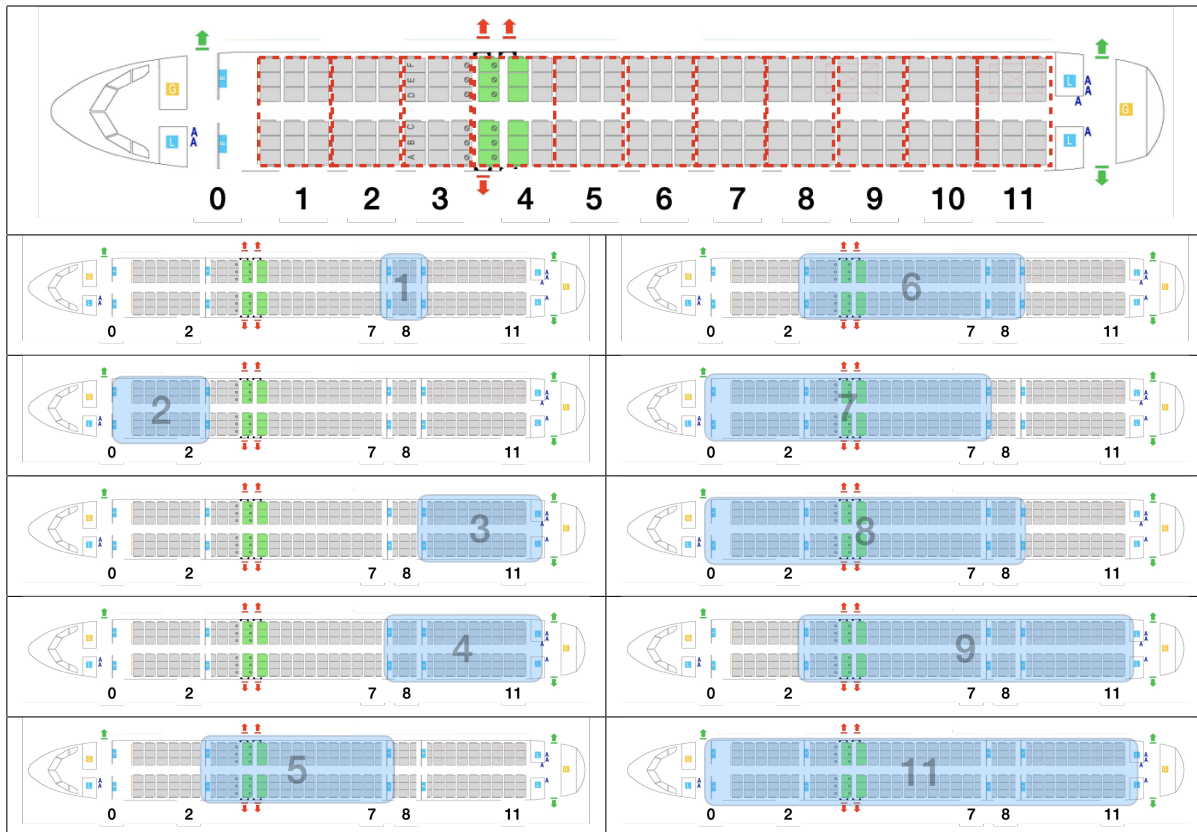


Figure 1: Exemple d'avion avec une capacité de 11 blocs, 5 diviseurs à installer et une issue de secours.

Question 1 • Proposez une solution sans Programmation par Contraintes et fournissez une implémentation en Java pour résoudre le problème des séparateurs de classes avec la fonction *dividers*($n, m, exits$).

Question 2 • Testez votre solution sur les instances fournies dans ce projet.

Question 3 • Maintenant, modélisez *dividers*($n, m, exits$) sous forme d'un réseau de contraintes (X, D, C).

Question 4 • Mettez en œuvre votre réseau de contraintes sous Choco en utilisant les fichiers fournis dans ce projet tout en respectant les principes SOLID.

Question 5 • Testez la version Choco sur les instances fournies dans le projet.

Question 6 • Optimisez votre modèle en expliquant en détail les améliorations apportées, telles que l'utilisation de contraintes globales, l'élimination des symétries, et la réduction des contraintes redondantes.