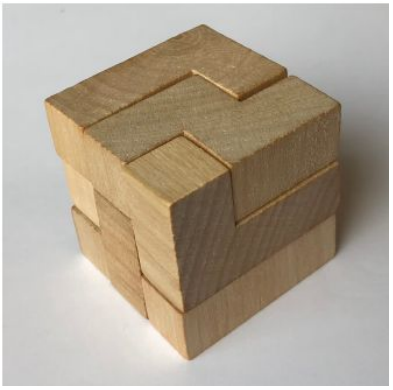
Rapport ASD1 (2) : Cube magique

Jonathan Zaehringer

Florian Schaufelberger

Jorge-André Fulgencio Esteves

Introduction



Le but de ce laboratoire est d'analyser le jeu illustré ci-contre. Il consiste en 7 pièces de bois en forme de C (1), de S (1), de T (1) et de L (4). Le coin occupe le volume de petits cubes 1x1x1. Les autres pièces celui de 4 petits cubes. Le but de jeu est de positionner ces 7 pièces de sorte à former un cube 3x3x3.

Démarche

Pour la réalisation de ce projet, nous avons décidé de poser de manière static l'ensemble des rotations des pièces en les collants au niveau du point (0,0,0) dans le plan $\mathbb{Z}^3$. Puis nous translatons l'ensemble de ces rotations dans le cube 3x3x3 pour avoir l'ensemble des positions possibles des pièces dans le cube. Notre solution est orientée objet avec héritage pour réduire les doublons de fonction. Nous avons un ensemble de pièce (C,S,T,L) qui hérite de la classe pièce qui contient la forme stocké par l'objet et un masque binaire. Nous travaillons avec des masques binaires pour accélérer les calculs. Le principe est que chaques rotations et translations d'une pièce est stockée sur un mask de 27 bits d'un entier, nous le calculons comme ceci :

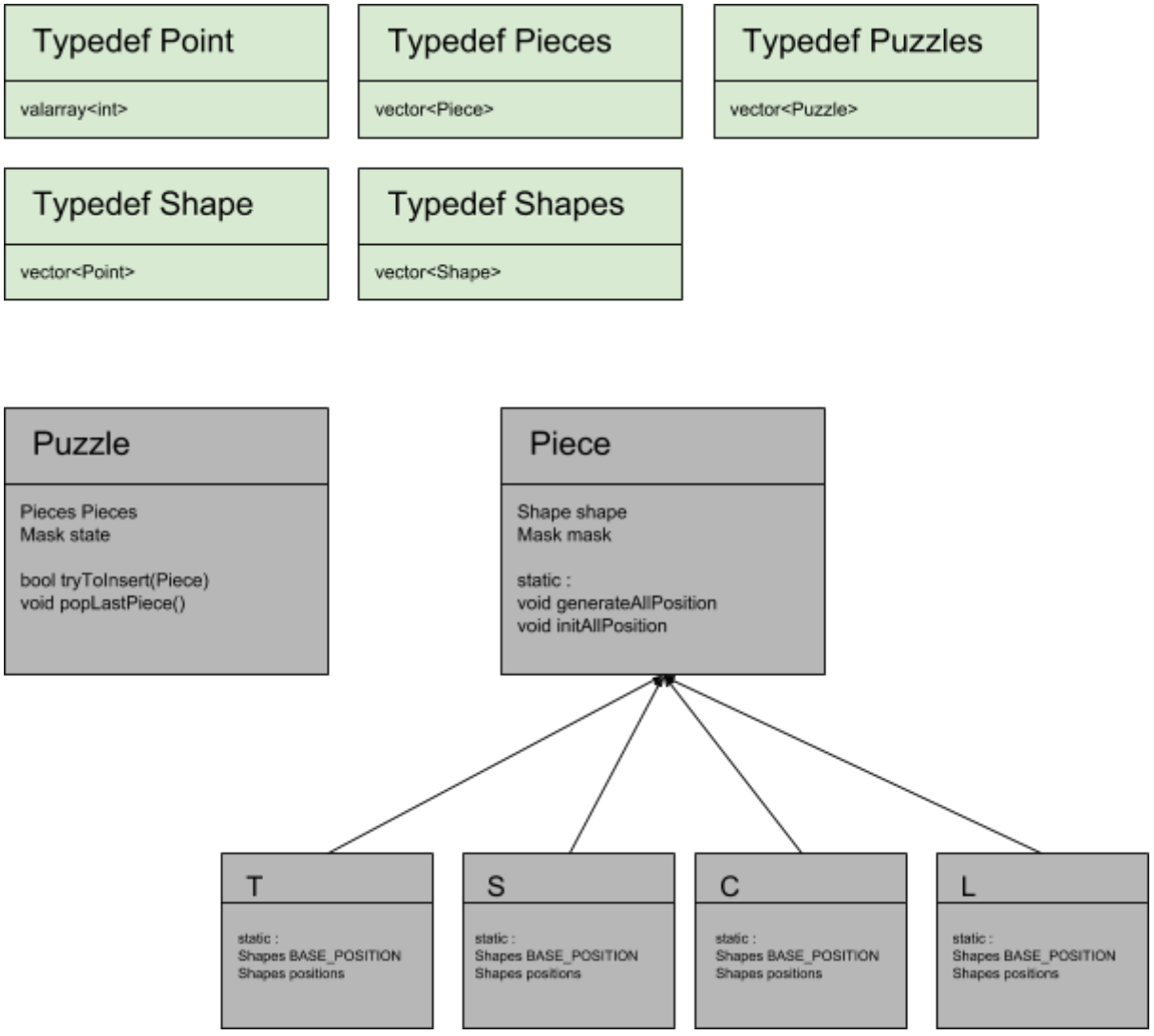
2^i	26	25	24	[...]	11	10	9	8	7	6	5	4	3	2	1	0
x	2	1	0	-	2	1	0	2	1	0	2	1	0	2	1	0
y	2	2	2	-	0	0	0	2	2	2	1	1	1	0	0	0
z	2	2	2	-	1	1	1	0	0	0	0	0	0	0	0	0

Ce masque nous permets de gérer l'insertion et la suppression d'une pièce dans le casse-tête. Pour ce faire, nous utilisons une classe puzzle qui permet de stocker les pièces insérées dans le casse-tête lors du brute force. Par un simple AND logique bit à bit de l'état du casse-tête et de la pièce a inséré, nous savons si cela est possible. Lors de la suppression, nous avons simplement besoin de soustraire le masque de la pièce à l'état du puzzle.

Jonathan Zaehringer
Florian Schaufelberger
Jorge-André Fulgencio Esteves

Pour le brute force des solutions, nous utilisons la fonction tryToInsert de puzzle qui permet d'être le premier cas trivial de la récursivité qui s'arrête lorsque que la solution n'est pas valable avec la pièce en cours. Le second cas trivial est lorsque nous atteignons le fond de l'arbre qui est la profondeur maximale, dès lors, nous connaissons une nouvelle solution au problème. Nous effectuons la suppression des doublons par permutation des L par la vérification qu'une solution est la même si les masques sont identiques.

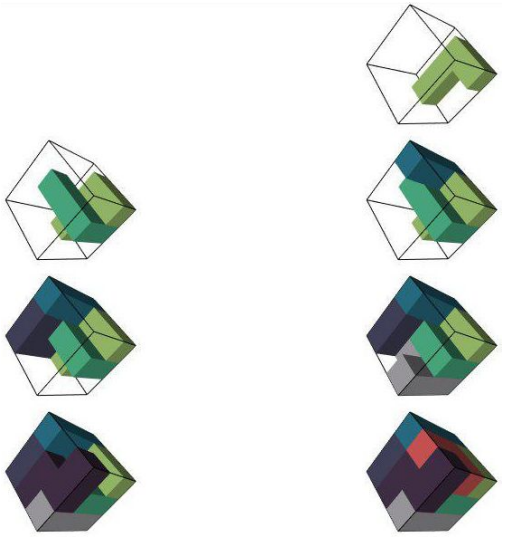
Simple diagramme de la structure du code.

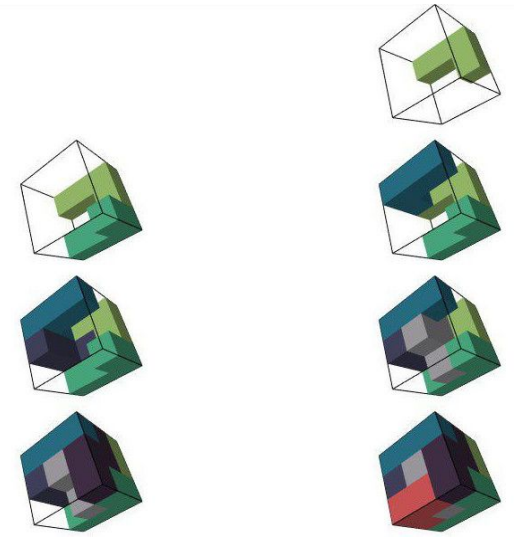


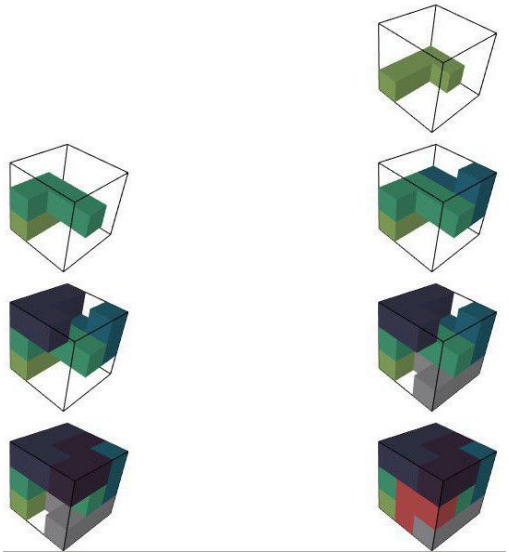
Jonathan Zaehringer
Florian Schaufelberger
Jorge-André Fulgencio Esteves

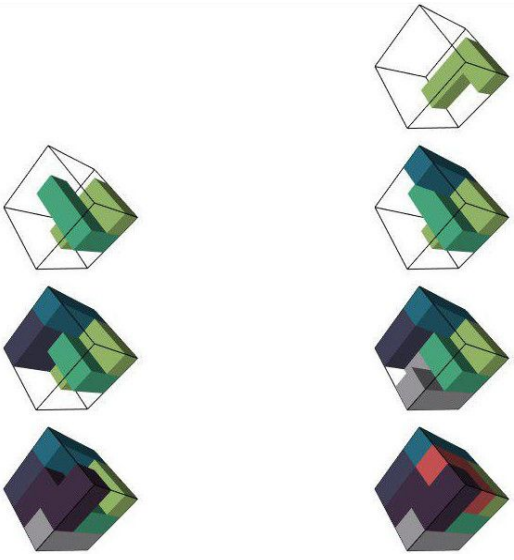
Réponses aux questions

-Donnez 4 solutions au jeu, qui ne sont pas équivalentes par rotation / par symétrie ou par échange de pièces identiques (deux L)









-Combien de solutions différentes y-a-t-il en tout ?

Il y a en tout 324’864 solutions.

Démarche de résolution

Nous avons brute force l’ensemble des possibilités avec notre programme pour trouver ce nombre. Toutefois nous avons voulu accélérer le temps de calcul avec différents principes. Pour commencer, nous avons fixé la pièce C sur une seule position de base car nous avons pu voir que cette pièce restreint le plus le nombre de solution possible ce qui permet d’arrêter plus rapidement le calcul d’une branche de l’arbre. De plus, il est important de placer le C en premier puis le T et le S pour maximiser le temps de calcul, la complexité de ces pièces engendrant une condition plus rapide dans la plupart des récursions. Le résultat obtenu étant 27’072 solutions. Par une multiplication par 12 dû à la simplification des positions du C nous arrivons au 324’864.

-Combien de solution différentes y-a-t-il si deux solutions ne différant que par l’échange de deux pièces L sont considérées comme une seule?

Il y a en 13’356 solutions sans permutations des L.

Démarche de résolution

Nous reprenons le nombre de solutions totales que nous avons trouvé avant et le divisons par 4 factoriel car cela enlève tout les répétitions dans les permutations dû aux 4 L. Ce qui nous donnera 324’864 / 4! = 13’536 solutions. Cette solution théorique étant validé par la pratique et l’application de notre brute-force en divisant le chiffre 27072 par deux pour supprimer la symétrie du C.

-Combien de solution différentes y-a-t-il si deux solutions ne différant que par rotation du cube sont considérées comme une seule?

Il y a en 27’072 solutions en enlevant la rotations du cube. Mais pas la symétrie du C ce qui donne au total 13’536.

Démarche de résolution

Pour enlever toutes les rotations nous devons diviser par 12 car nous avons 12 fois trop de solutions à cause des rotations du cube par la pièce C. Ce qui nous donne 324’864 / 12 = 27’072 solutions si on enlèves les rotations sans la symétrie du C ce qui nécessite une division par deux soit 13’536 solutions.

Jonathan Zaehringer
Florian Schaufelberger
Jorge-André Fulgencio Esteves

-Combien de solution différentes y-a-t-il si ces deux critères sont appliqués?

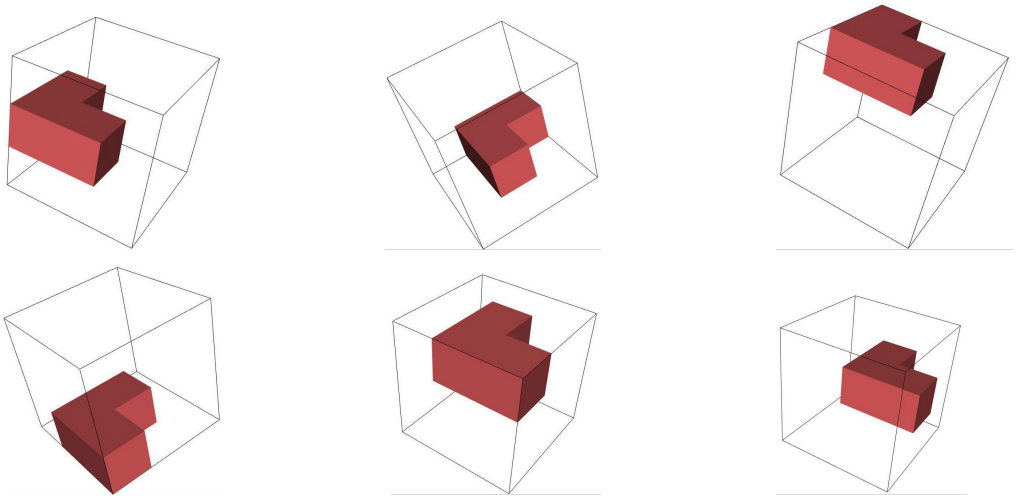
Nous obtenons 1'128 solutions sans prendre en compte la symétrie du la pièce C ce qui donne au final 564 solutions uniques

Démarche de résolution

La solution a pu être brute force par notre code sans prendre en compte la symétrie du C. Ce qui valide l'ensemble des questions précédentes.

-Si l'on place d'abord la pièce en coin, y-a-t-il des positions qui rendent impossible toute solution? Si oui, lesquelles?

Oui il y a des position qui rendent impossible toutes solutions. Voici les 6 positions qui nous empêche de trouver des solutions.



Démarche de résolution

Pour trouver, nous avons pris notre vecteur contenant les positions possible de notre pièce C et nous avons comparé avec nos solutions pour compter le nombre de solution dans lequel la pièce apparaissait à cette position si le compteur retournait 0 cela implique que la position n'engendre aucune solution dans le cube 3x3x3.

Jonathan Zaehringer
Florian Schaufelberger
Jorge-André Fulgencio Esteves

-Quelles autres combinaison de 7 pièces (parmi coin, S, T et L) permettent de réaliser un cube 3x3x3? Lesquelles ne le permettent pas?

21 solutions permettent de réaliser le casse-tête 3x3x3 qui sont les suivantes :

{C, T, T, T, T, T, L}	{C, T, T, T, T, S, L}	{C, T, S, S, S, S, L}	{C, S, S, S, S, L, L}
{C, T, T, S, S, S, L}	{C, T, T, T, T, L, L}	{C, T, S, S, S, L, L}	{C, S, S, S, L, L, L}
{C, T, T, S, S, L, L}	{C, T, T, T, S, S, S}	{C, T, S, S, L, L, L}	{C, S, S, L, L, L, L}
{C, T, T, S, L, L, L}	{C, T, T, T, S, S, L}	{C, T, S, L, L, L, L}	{C, S, L, L, L, L, L}
{C, T, T, L, L, L, L}	{C, T, T, T, S, L, L}	{C, T, L, L, L, L, L}	{C, L, L, L, L, L, L}
{C, T, T, T, L, L, L}			

Toutes les solutions contenant aucune ou plus de une pièce C ainsi que les 7 solutions explicitées ci-dessous ne permettent pas de résoudre un 3x3x3.

{C, T, T, T, T, T, T}	{C, T, T, T, T, S, S}	{C, T, S, S, S, S, S}	{C, S, S, S, S, S, L}
{C, T, T, T, T, T, S}	{C, T, T, S, S, S, S}	{C, S, S, S, S, S, S}	

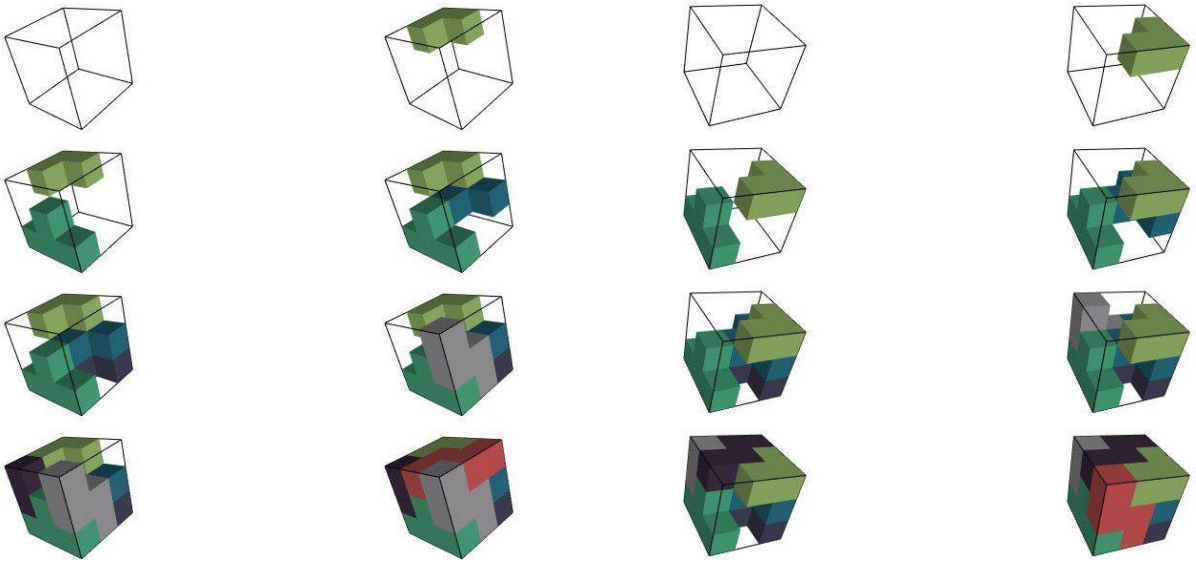
Démarche de résolution

Pour répondre à cette question, nous avons dans un premier temps cherché quelles combinaisons de 7 pièces parmis {C, S, T, L} peuvent rentrer dans le cube. Pour ce faire nous avons compté le nombre de cube unitaire lors de la construction de notre combinaison et si elle atteint 27 à 7 pièces, cette combinaison est potentiellement plausible. Avec les 28 solutions trouvées, nous “brute-forçons” ces combinaison successivement pour obtenir le nombre de solutions.

Jonathan Zaehringer
Florian Schaufelberger
Jorge-André Fulgencio Esteves

-Parmi les combinaisons de 7 pièces différentes permettent de réaliser un cube 3x3, laquelle offre le moins de solutions différentes?

La combinaison qui propose le moins de solutions est la suivante : {C, T, T, T, S, S, S}. En effet elle n'a que 2 solutions sans rotation et sans échange entre les pièces. Les solutions du cube avec lesdites pièce sont :



1ère solution *2ème solution*

Démarche de résolution

Une fois toutes les solutions générées pour les combinaisons probables, il nous a suffit lors du parcours du vecteur de solutions de stocker la solution engendrant le moins de solutions (et > 0). Il nous aura fallu diviser le résultat par deux car il y a une symétrie dans la disposition de la pièce C.

Conclusion

Ce labo était très intéressant au niveau de la conception du problème et du challenge proposé , mais la récursivité n'a été qu'une très faible partie de la complexité de ce labo. La conception et l'optimisation de notre code étant ce qui nous a posé le plus de problème. Le challenge donné par ce labo étant plus porté sur la partie stockage de l'information et son traitement que sur une logique de récursivité. Ce laboratoire s'est avéré être très chronophage mais resta néanmoins amusant que ce soit par le sujet que par la collaboration avec un élève de 3ème année.