| *Novembre 2023* | *Université de Paris 8* |
| --- | --- |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Projet Tuteurés 2023-2024

Pavages de Polyominos en JS

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Viet NGUYEN

Damien GEOFFROY

# Table des matières

[Table des matières 2](#_4d34og8)

[Remerciements 3](#_17dp8vu)

[I. Introduction 3](#_3rdcrjn)

[II. Exploration des Types de Polyomino en 2D 4](#_26in1rg)

[Introduction aux Polyomino 4](#_lnxbz9)

[Description du pavage 5](#_35nkun2)

[Règle de classification des Polyominos 5](#_1ksv4uv)

[Tableau des pièces de Polyominos 6](#_44sinio)

[Cas particuliers 7](#_z337ya)

[III. Les Polycubes 8](#_3j2qqm3)

[Présentation des Polycubes 9](#_1y810tw)

[Caractéristiques des Polycubes 9](#_4i7ojhp)

[Challenges et Complexités liés aux Polycubes 10](#_2xcytpi)

[IV . Utilisation de JavaScript dans la réalisation du pavage de polyominos 11](#_1ci93xb)

[Introduction à JavaScript 11](#_3whwml4)

[Présentation brève 11](#_2bn6wsx)

[Pertinence de JavaScript pour le projet 12](#_qsh70q)

[Capacités Graphiques de JavaScript pour la Visualisation 12](#_3as4poj)

[Bibliothèques graphiques populaires en JavaScript 12](#_1pxezwc)

[V. Concept 14](#_49x2ik5)

[Décrire le fonctionnement Pavages de Polyominos (Polyomino Tilings) 14](#_2p2csry)

[Les algorithmes importants 15](#_147n2zr)

[Conception de l'interface utilisateur 16](#_3o7alnk)

[Version 2D 16](#_23ckvvd)

[Identification des structures de données 16](#_ihv636)

[Évaluer la fonctionnalité et la faisabilité de la conception 17](#_32hioqz)

[VI. Développement (Sera mis à jour à l'avenir) 18](#_1hmsyys)

[Déployer le code source et créer des fonctions selon les besoins 18](#_41mghml)

[Vérifier et corriger les erreurs 18](#_2grqrue)

[Assurer le respect des normes de programmation et des règles d'exécution des projets 18](#_vx1227)

[Signaler les progrès et les problèmes rencontrés 18](#_3fwokq0)

[VII. Tests et validation (Sera mis à jour à l'avenir) 18](#_1v1yuxt)

[VIII. Documentation (Sera mis à jour à l'avenir) 18](#_4f1mdlm)

[IX. Formation (Sera mis à jour à l'avenir) 19](#_2u6wntf)

[X. Livraison (Sera mis à jour à l'avenir) 19](#_19c6y18)

[XI. Maintenance et assistance (Sera mis à jour à l'avenir) 19](#_3tbugp1)

[XII. Conclusion 19](#_28h4qwu)

[XIII. Annexes (Sera mis à jour à l'avenir) 20](#_nmf14n)

[XIV. Bibliographie 20](#_37m2jsg)

[Sources utilisé pour les images et les notes 20](#_1mrcu09)

[Sources intéressantes dans le réalisation de notre document 20](#_46r0co2)

# **Remerciements**

Nous remercions Monsieur Nicolas JOUANDEAU pour son accompagnement et ses conseils qui nous ont permis de réaliser ce document et d'appréhender ce projet tutoré. Nous exprimons également notre gratitude envers Monsieur Farès BELHADJ pour avoir mis à notre disposition son document, nous permettant ainsi de fournir un rendu professionnel.

# **I. Introduction**

Ce document PDF représente un état de l'art approfondi sur le sujet des pavages de polyominos. Cette étude s'inscrit dans le cadre de notre projet tuteuré, mené au sein de notre cursus à l’Université Paris 8 Vincennes - Saint-Denis, pendant notre troisième année. Ce projet tuteuré se décline en deux phases distinctes réparties sur les deux semestres. La première phase consiste en une recherche et réflexion approfondie, aboutissant à la rédaction de cet état de l'art que vous êtes en train de lire. La seconde partie de notre projet implique la mise en pratique des connaissances acquises, avec la concrétisation du projet dans son ensemble.

Le sujet qui nous a été attribué est particulièrement captivant. Il nous offre l'opportunité d'explorer les domaines des mathématiques à travers l'étude des polyominos. Ce sont des formes géométriques, constituées de l'assemblage de carrés unitaires connectés entre eux, pouvant varier en taille allant d’une taille 1 à une limite indéterminée. Les polyominos ont gagné en popularité au cours du 20e siècle, tant en raison des problèmes mathématiques complexes qu'ils suscitent que grâce à la diffusion mondiale du jeu Tetris. Ce dernier, basé sur des pièces de polyominos tombant sur une grille de dimensions finies, a contribué à populariser ces formes simples. Le jeu consiste à compléter des lignes en largeur pour augmenter le score du joueur.

Ce projet nous offre une perspective unique en nous permettant d'explorer non seulement le domaine des mathématiques, grâce au pavage qui consiste en une disposition régulière d’éléments les uns à-côtés des autres afin d’obtenir une nouvelle forme. Dans le cadre de notre sujet, il s’agit de l'emboîtement des polyominos, mais également celui de l'informatique et de l'algorithmique. Nous mettrons en œuvre nos connaissances, notamment en utilisant JavaScript, un langage de programmation offrant des capacités graphiques avancées tout tout en restant flexible. Sa popularité et sa compatibilité avec les navigateurs web en font un outil idéal pour la visualisation des résultats des algorithmes de pavage. De plus, la syntaxe claire et la facilité de manipulation des objets dans JavaScript faciliteront le processus de conception et de mise en œuvre de notre programme.

Tout au long de ce document, nous allons plonger dans les aspects théoriques des pavages de polyominos, en préparation à la phase pratique de notre projet tuteuré dans laquelle nous exploiteront pleinement les fonctionnalités de JavaScript pour concrétiser nos idées et visualiser les résultats obtenus.

# **II. Exploration des Types de Polyomino en 2D**

Dans ce chapitre, nous explorerons les divers polyominos existants, offrant ainsi une compréhension approfondie de l'enjeu de notre projet tuteuré. Pour ce faire, nous reverrons les fondements même d'un polyomino, cela nous amènera à voir leurs caractéristiques essentielles, tout en élargissant notre analyse pour examiner les nombreuses variantes qui existent pour certains polyominos spécifiques. Cette démarche nous permettra de comprendre les défis et les problèmes liés à ces derniers, nous permettant d’avoir un point de vue global des possibilités et des complexités que nous devrons aborder dans la réalisation de notre projet.

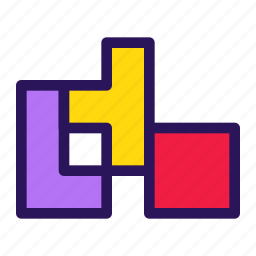
## **Introduction aux Polyomino**

Avant de poursuivre notre étude, revenons sur la nature même d'un polyomino.

Le terme "polyomino" a été introduit en 1953 par Solomon W. Golomb (Source : [Solomon W. Golomb - Wikipedia](https://en.wikipedia.org/wiki/Solomon_W._Golomb) ), un ingénieur et mathématicien célèbre pour ses contributions variées, dont l'invention des "échecodames" [[1]](#footnote-0) et son empreinte laissée grâce à ses travaux sur les polyominos.

L'appellation "polyomino" est dérivée du mot "domino", et le préfixe "poly" peut être substitué par le nombre de parties composant le polyomino en grec. Un polyomino, tel que décrit précédemment, se compose d'un ensemble de carrés connectés entre eux par les bords, le différenciant des autres polyformes tels que ceux formés de répétition pyramides ou des cubes.

Dans le cadre de notre projet tuteuré, nous focalisons notre attention sur les polyominos en raison de leur prédominance et de leur utilisation évidente dans le contexte d’un plan (leurs équivalents tridimensionnels étant les polycubes). En effet, travailler avec des polyominos présente des avantages pratiques, notamment sur le plan 2D. Par rapport aux polyiamondes, qui ont une forme de base triangulaire, les polyominos offrent une plus grande simplicité, réduisant les complications liées aux espaces perdus engendrés par cette forme particulière. D'un autre côté, les formes plus rectangulaires s’avèrent être des équivalents aux polyominos, rendant leur étude moins captivante dans le cadre de notre projet tuteuré.

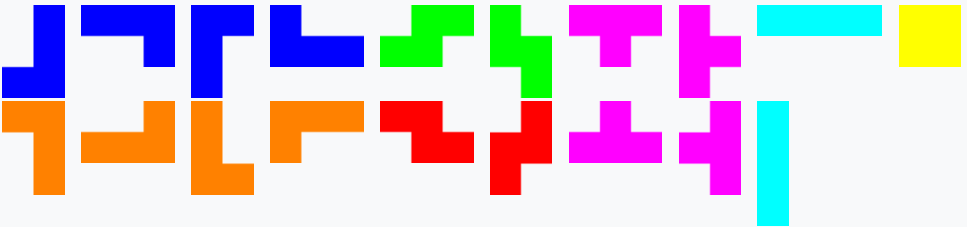


## **Description du pavage**

### **Règle de classification des Polyominos**

Lorsqu'il s'agit de trier les polyominos, une règle de classification en trois catégories est communément utilisée.

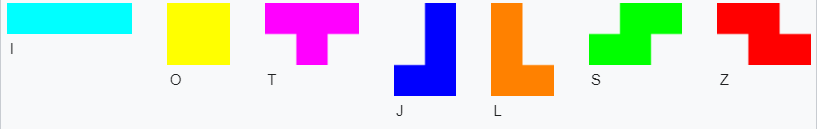
Tout d'abord, nous avons les polyominos dits "libres". Il s'agit de l'ensemble des polyominos différents, séparés en rotation, réflexion ou translation. Par conséquent, les tétraminos de type "Z" et "S" et “N” sont distingués en fonction de leur réflexion et rotation respective.



*Figure montrant les tetromino de manière fixe ,*

*les formes ont été réarrangé par nous et viennent de :* [*Tetromino - Wikipedia*](https://en.wikipedia.org/wiki/Tetromino)

La seconde classification, plus restrictive, est appelée "unilatérale". Elle exclut la réflexion, ne conservant que les polyominos en fonction de leur rotation et de leur translation. Cette classification offre une vision plus restreinte mais parfois plus pertinente selon le contexte d'application.



*Figure montrant les tetromino de manière unilatérale , les formes ont été réarrangé par nous et viennent de :* [*Tetromino - Wikipedia*](https://en.wikipedia.org/wiki/Tetromino)

Enfin, la classification dite "fixée" se base uniquement sur la forme des polyominos, sans prendre en compte la réflexion, la rotation ou la translation. Cette approche, moins flexible, est très pratique pour le dénombrement des différentes formes, et permet ainsi une perspective unique sur la variété des polyominos.



*Figure montrant les tetromino de manière unilatérale , les formes ont été réarrangé par nous et viennent de :* [*Tetromino - Wikipedia*](https://en.wikipedia.org/wiki/Tetromino)

La compréhension de ces différentes classifications est essentielle pour notre projet tuteuré, car elle influence directement la manière dont nous aborderons les différents types de polyominos dans notre exploration algorithmique.

### **Tableau des pièces de Polyominos**

Pour une explication plus fluide et simplifié voici un tableau récapitulant les polyominos en fonction des différentes types de classifications [[2]](#footnote-1)

### 

|  | **forme** | **total** | **avec trous** | **sans trous** | **unilatérale** | **fixe** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | monomino | 1 | 0 | 1 | 1 | 1 |
| 2 | domino | 1 | 0 | 1 | 1 | 2 |
| 3 | triomino | 2 | 0 | 2 | 2 | 6 |
| 4 | tetramino / quadriminos | 5 | 0 | 5 | 7 | 19 |
| 5 | pentamino | 12 | 0 | 12 | 18 | 63 |
| 6 | hexamino | 35 | 0 | 35 | 60 | 216 |
| 7 | heptamino | 108 | 1 | 107 | 196 | 760 |
| 8 | octamino | 369 | 6 | 363 | 704 | 2725 |
| 9 | nonominos/enneamino | 1285 | 37 | 1248 | 2500 | 9910 |
| 10 | decamino | 4655 | 195 | 4460 | 9189 | 36 446 |
| 11 | undecamino | 17073 | 979 | 16094 | 33896 | 135 268 |
| 12 | dodecamino | 63 600 | 4 663 | 58 937 | 126 759 | 505 861 |
| 15 | Pentédécominos | 3 426 576 | 424 056 | 3 002 520 |  |  |

"Ces données riches, présentées dans le tableau, nous fournissent des informations essentielles, mais surtout elles mettent en évidence la présence de certains cas particuliers que nous allons évoquer.

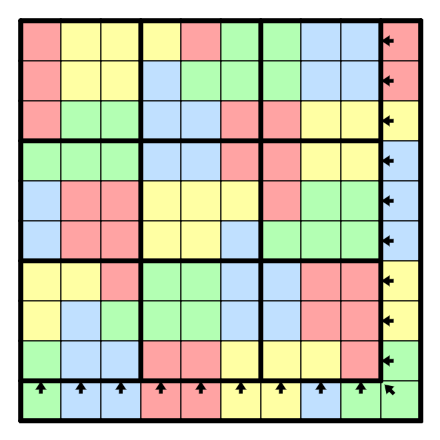
### **Cas particuliers**

La présence quasi infinie de polyominos rend impossible une étude exhaustive. Cependant, en examinant certains polyominos déjà analysés par des chercheurs, des cas particuliers se dégagent. Notamment, à partir des heptaminos, nous constatons la présence de pièces comportant des trous. Ces configurations, par définition, empêchent la réalisation d'un pavage parfait et, par extension, intensifient la complexité du pavage.

*Figure montrant les polyominos de 7 à 9 carrés* 

*avec des trous :* [Polyomino -- from Wolfram MathWorld](https://mathworld.wolfram.com/Polyomino.html)

Une singularité supplémentaire réside dans le fait que certains types de pavages sont purement et simplement impossibles à remplir et à construire en fonction des formes choisies. Cette démonstration peut être réalisée par des calculs mathématiques (par exemple un jeu de plateau de dame composé de 64 cases ne peut être rempli avec des triominos car 64 n’est pas divisible par 3)[[3]](#footnote-2) ou simplement par une logique claire, comme illustré dans la figure suivante :

*Figure montrant l’impossibilité de remplir un rectangle de 9*

*avec des polyomino différent du mono, trio et nonominos :* [logic-masters.de](https://logic-masters.de/Raetselportal/Raetsel/zeigen.php?chlang=en&id=00041R)

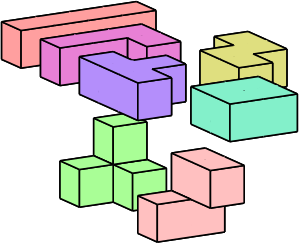
Par ailleurs, les polyominos de taille importante peuvent poser des défis en raison du nombre limité d'options disponibles pour les pavoirs, complexifiant ainsi la recherche de solutions.

Un autre cas particulier intéressant concerne la présence de motifs dits auto-répliquant, créant ainsi des schémas répétitifs fascinants. Ces motifs offrent des perspectives uniques et ajoutent une dimension esthétique à l'étude des polyominos. Nous pouvons citer par exemple deux dominos coller entre-eux formant un carré de taille 2x2.

Ces cas particuliers enrichissent la compréhension des polyominos pour les pavages, et leur prise en compte peut considérablement influencer la complexité des problèmes et des algorithmes associés. Ils constituent des défis intéressants pour la résolution de problèmes mathématiques et informatiques liés aux polyominos et c’est en gardant ces nuances à l'esprit tout au long de notre analyse que nous serons en mesure de concrétiser le projet lors du second semestre.

# **III. Les Polycubes**

Dans le chapitre précédent, nous avons exploré les polyominos en deux dimensions, découvrant leurs propriétés, classifications et leurs caractéristiques dans les pavages 2D. À présent, voyons leurs équivalents lorsque l’on ajoute une dimension supplémentaire à savoir les polycubes en 3D. Dans ce chapitre, nous verrons la définition des polycubes, leurs classifications, ainsi que leurs caractéristiques. En comparant avec les polyominos, nous mettrons en évidence les différences spécifiques aux pavages en trois dimensions.

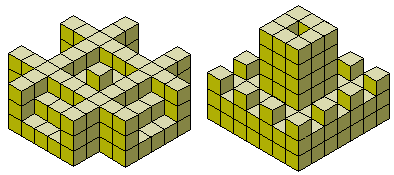


## **Présentation des Polycubes**

Comme nous l’avions mentionné précédemment, le terme ‘polycubes’ suit la même logique d'appellation que les polyominos. Ici, le préfixe 'poly' est remplacé par le nombre d'éléments, et le suffixe ‘cube’ lui y est ajouté. Les polycubes représentent l'équivalent tridimensionnel des polyominos. Contrairement à ces derniers, les polycubes sont formés par des cubes unitaires connectés les uns aux autres au lieu de carrés. Cette simple modification engendre une transformation significative, introduisant un nouvel axe et permettant une diversité bien plus grande de formations. Ainsi, l'ensemble des polycubes s'avère plus vaste que celui des polyominos, ouvrant la porte à une variété accrue dans la construction d'ensembles tridimensionnels.

## **Caractéristiques des Polycubes**

Le nouveau plan ajouté permet une complexité bien plus importante, avec des formes pouvant être plus ou moins profondes. De nouvelles constructions, considérées comme impossibles, deviennent envisageables notamment grâce à la manipulation des connexions entre les cubes pour insérer plus de trous. Malgré tout tant qu'une face d'un cube est connectée à une autre face, la formation est considérée comme valide donc aucun cube ne peut être uniquement relié par une arête comme illustré dans ces exemples :

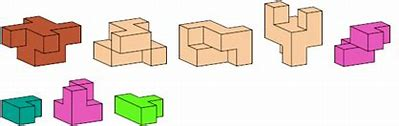


*Figure montrant des figures complexe de polycube :* [logic-masters.de](https://logic-masters.de/Raetselportal/Raetsel/zeigen.php?chlang=en&id=00041R)

Nous remarquons également que les règles de classification des polycubes sont identiques à celles des polyominos. Ainsi, nous avons des polycubes "fixes", "libres" et "unilatéraux".

Cependant, leurs points distinctifs sont dans leurs symétries qui sont plus nombreuses et potentiellement plus voyantes. Nous pouvons également voir que certains polycubes ne peuvent pas être retournés ou réfléchis comme les polyominos, et la plupart sont asymétriques. Voici une liste des polycubes avec leurs nombres respectifs[[4]](#footnote-3) :

|  | **forme** | **Nombre de n-polycubes**  **unilatéraux (les réflexions**  **sont comptées comme istinctes)** | **Nombre de n-polycubes**  **libres (les réflexions**  **sont comptées ensemble)** |
| --- | --- | --- | --- |
| 1 | monocube | 1 | 1 |
| 2 | dicube | 1 | 1 |
| 3 | tricube | 2 | 2 |
| 4 | tetracube | 8 | 7 |
| 5 | pentacube | 29 | 23 |
| 6 | hexacube | 166 | 112 |
| 7 | heptacube | 1023 | 607 |
| 8 | octocube | 6922 | 3811 |



## **Challenges et Complexités liés aux Polycubes**

La connexion entre les cubes étant plus complexe que pour les polyominos, cela nécessite la création d'algorithmes capables de comprendre un espace tridimensionnel plus dynamique. Trouver la disposition la plus efficace tout en respectant les contraintes spécifiques des polycubes, telles que les limitations de taille, de forme ou de connectivité dynamique entre les cubes imposées par les polycubes eux-mêmes, ajoute une dose supplémentaire de complexité. Une approche algorithmique sophistiquée est donc requise, notamment pour éviter les collisions entre les formes et maximiser la comblage des trous pour un pavage parfait.

Malgré ces défis, les polycubes offrent des opportunités passionnantes dans la modélisation 3D de notre projet, augmentant la réflexion nécessaire à la résolution des problèmes de pavage.

# **IV . Utilisation de JavaScript dans la réalisation du pavage de polyominos**

Dans ce chapitre nous présenterons Javascript dans le contexte de notre projet. Après une brève introduction à JavaScript, nous allons voir ses capacités graphiques et sa flexibilité algorithmique, soulignant son rôle essentiel dans la visualisation et la manipulation de données pour ce type de problème.Pour cela, nous discutons du rôle de JavaScript dans notre projet tuteuré de pavage de polyominos, en mettant en évidence ses avantages et limitations. Enfin , nous allons voir des exemples concrets de mise en œuvre en 2D et 3D, illustrant comment JavaScript peut être employé pour résoudre des défis spécifiques de pavage.

## **Introduction à JavaScript**

### **Présentation brève**

JavaScript, souvent abrégé en JS (à ne pas confondre avec Java), est un langage de programmation créé en 1995. Fréquemment utilisé avec HTML et CSS, JavaScript joue un rôle essentiel dans la création de pages web dynamiques en permettant la manipulation des données. En tant que langage orienté objet, il utilise des prototypes, offrant une structure flexible de variables et de méthodes réutilisables à travers le code. Cette approche facilite la transmission des données et permet des modifications tant du côté client que du côté serveur.

Du côté client, JavaScript intervient en modifiant le Document Object Model (DOM), la structure de la page HTML, en ajoutant des éléments de manière dynamique. Du côté serveur, il peut être employé pour manipuler des bases de données, par exemple.

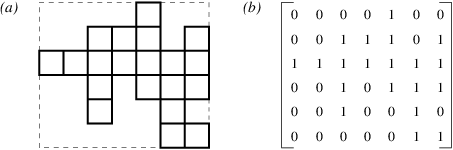
### **Pertinence de JavaScript pour le projet**

Dans le cadre de notre projet tuteuré, l'utilisation du langage JavaScript est imposée, mais nous considérons ce choix comme particulièrement approprié. Cette pertinence réside notamment dans la gestion graphique offerte par JavaScript, que nous détaillerons plus ultérieurement. La facilité d'utilisation de bibliothèques pour obtenir des résultats en 2D et 3D est un atout majeur de JavaScript, une capacité que tous les langages ne possèdent pas, ou qui nécessite des approches plus complexes. De plus, couplé à cela, JavaScript se distingue par sa capacité à gérer des algorithmes complexes, offrant ainsi une solution complète et adaptée à notre projet de pavage de polyominos.

## **Capacités Graphiques de JavaScript pour la Visualisation**

### **Bibliothèques graphiques populaires en JavaScript**

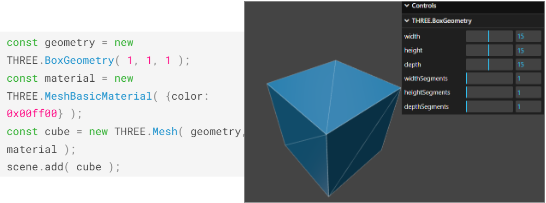
Pour élargir les possibilités de manipulation graphique en JavaScript, l'utilisation de ce que nous appelons des "*bibliothèques*" s'avère essentielle. Ces bibliothèques, constituées de fichiers contenant des ensembles de fonctions, simplifient énormément la réalisation de nombreuses opérations. Elles sont d’une importance particulière car elles permettent d'accomplir des tâches qui ne sont pas normalement disponibles avec le JavaScript natif, ou du moins pas aussi facilement. Dans le cadre de notre projet tuteuré, nous devons nous baser sur la librairie Three.js et l'API Canvas. Ces deux outils sont d'une importance capitale pour la présentation et la compréhension des solutions de pavage. Bien que des alternatives existent dans le domaine de la 2D, il est presque impossible d'obtenir des solutions équivalentes en 3D.



*Figure montrant une représentation textuel (ici dans une matrice) des polyominos :*

[A-polyomino-and-its-representation-as-a-binary-picture-or-matrix\_fig1\_268167195](https://www.researchgate.net/figure/A-polyomino-and-its-representation-as-a-binary-picture-or-matrix_fig1_268167195)

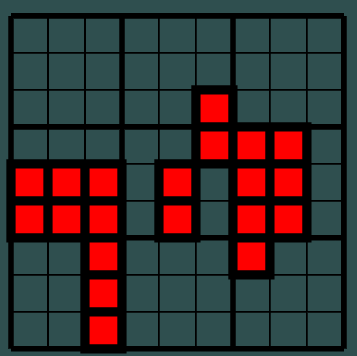
Pour l'instant, concentrons-nous sur l'aspect 3D avec Three.js. Cette bibliothèque facilite la création d'objets graphiques en permettant la mise en place d'une scène. À partir de cette scène, il devient possible d'ajouter une caméra pour obtenir un point de vue défini, ainsi qu'une source lumineuse.

*Figure montrant la création d’un cube en Three.js :* [*BoxGeometry – threejs.doc*](about:blank)

Si la création d'un cube est relativement facile, la gestion dynamique de ces objets et leur création dynamique sont bien plus complexes. Three.js offre un ensemble d'outils pour relever ces défis, mais la maîtrise de ces fonctionnalités demande une compréhension approfondie.

Passons à l'aspect 2D, pour cela nous ferons usage de l'API Canvas, il s’agit d’une interface de programmation permettant de dessiner et de manipuler des graphiques en 2D sur une page web, de manière dynamique si nécessaire. L'API Canvas met à disposition un objet de contexte 2D, obtenu grâce à la méthode ***getContext('2d')***. Cela permet de dessiner diverses formes telles que des rectangles, des cercles et des lignes, en plus de permettre la modification de leurs couleurs.

Il est donc possible à partir de l'utilisation de ces bibliothèques de nous rapprocher de certains autres projets basés sur les polyominos pour notre projet tuteuré, comme illustré par les images ci-dessous.



*Figure montrant un projet basé sur polyomino en Javascript*

*(cependant il n’a pas le même objectif final) :* [*Polyomino Block*](https://omerkel.github.io/PolyominoBlocks/html5/src/index.html)

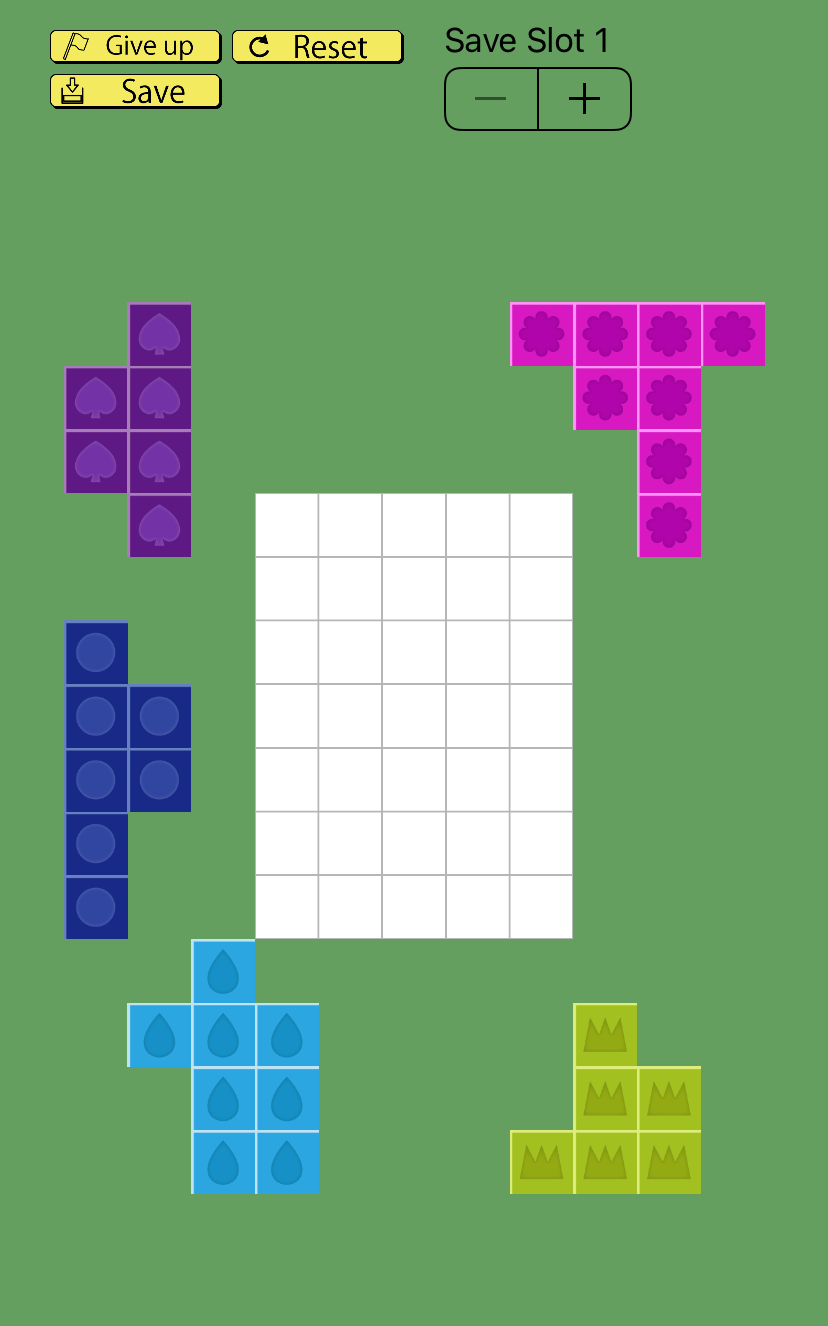
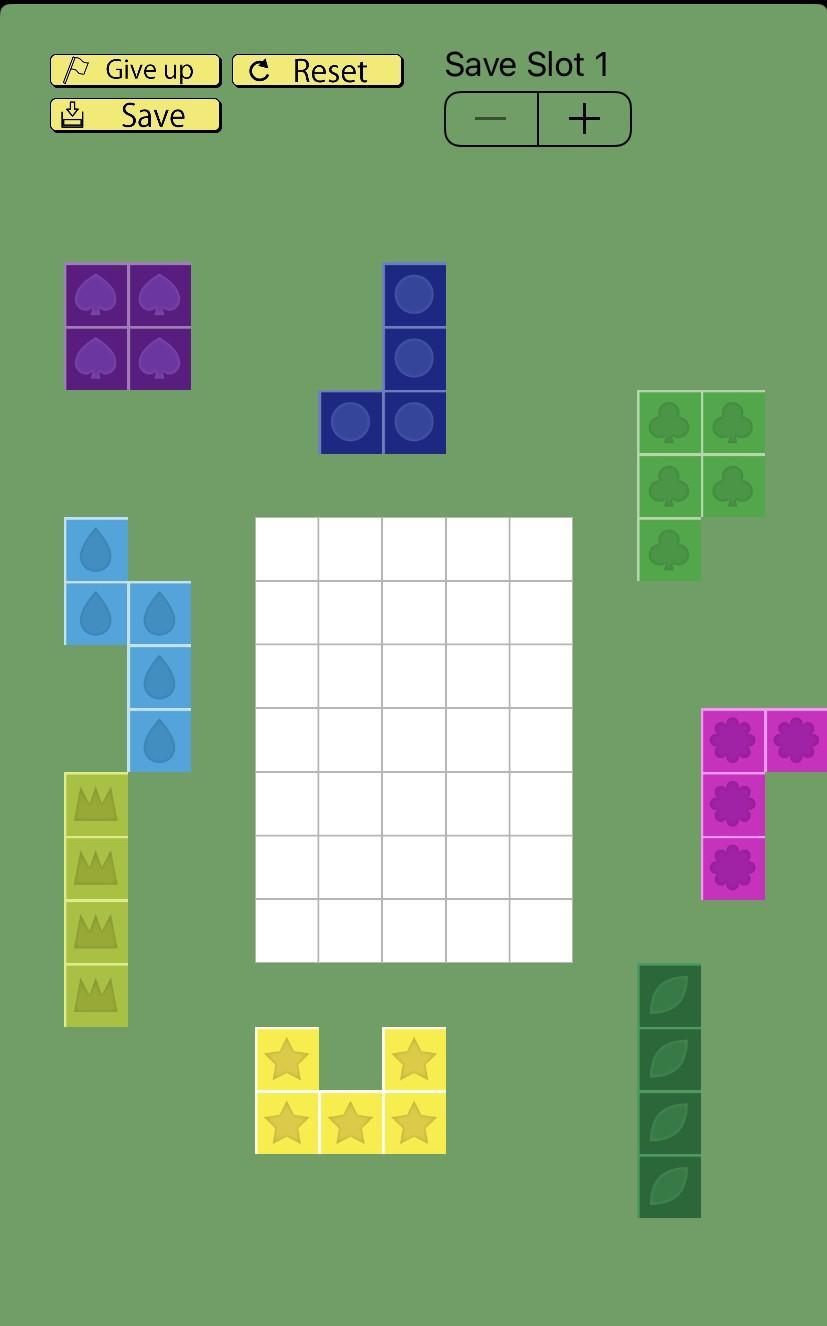
Comme nous pouvons le voir, ces outils offrent un ensemble diversifié pour aborder notre projet sous différents angles, que ce soit en explorant la complexité des formes en 3D ou en manipulant graphiquement des solutions de pavage en 2D.

# **V. Concept**

## **Décrire le fonctionnement Pavages de Polyominos (Polyomino Tilings)**

Dans le cadre du projet Pavages de Polyominos, nous explorons deux aspects distincts : le 2D et le 3D. Nous débutons par le 2D afin de décrire précisément le problème de l'agencement de polyominos sur une grille 2D sans laisser d'espaces vides, ce qui constitue le défi des Pavages de Polyominos. Il s'agit d'une catégorie de problèmes mathématiques et informatiques captivante. L'objectif consiste à placer un ensemble fini de polyominos (des pièces de formes rectangulaires) sur une grille 2D, sans laisser de vides ni de chevauchements entre les polyominos. Cette tâche exige l'élaboration de méthodes pour disposer les polyominos de manière à former une surface plane sans lacunes.

La disposition des pièces de Polyominos sur la grille, sans laisser d'espaces vides, rappelle la tâche courante de certains jeux de puzzle actuellement disponibles, tels que le jeu Jigsaw Polyomino Puzzle sur l'App Store :



*Figure Polyomino sur App Store*

Les Pavages de Polyominos suscitent un vif intérêt dans le domaine des mathématiques et constituent une part essentielle des jeux de logique et des casse-tête. Trouver des configurations de polyominos pour compléter un pavage (une couverture complète) sans laisser de vides représente un défi qui requiert une créativité mathématique.

Il est important de noter que les Pavages de Polyominos se distinguent des solveurs de Polyominos. Ces derniers englobent des outils ou des logiciels permettant de résoudre divers problèmes liés aux polyominos, y compris des tâches autres que l'agencement sur une grille 2D. Par exemple, un solveur de polyominos peut servir à créer des énigmes logiques ou à effectuer des calculs en rapport avec les polyominos, sans nécessairement impliquer la création d'un pavage.

En somme, notre projet se concentre sur les Pavages de Polyominos. Nous créerons une grille de carrés de taille personnalisée et résoudrons le défi des Pavages de Polyominos en disposant tous les carrés de la grille sans laisser de vides.

## **Les algorithmes importants**

Dans ce problème, l'objectif est de disposer un ensemble fini de polyominos (des pièces ayant des formes rectangulaires) sur une grille 2D sans laisser d'espaces vides et sans chevauchement entre les polyominos. Cette tâche requiert la recherche de méthodes pour organiser les polyominos de manière à former une surface plane sans laisser de lacunes.

Le recours à plusieurs algorithmes et méthodes est essentiel pour résoudre les Pavages de Polyominos, notamment :

* L'algorithme de recherche exhaustive (Backtracking) : Il explore toutes les possibilités de placement des polyominos et vérifie leur validité, en revenant en arrière en cas d'échec.
* La méthode Branch and Bound : Cette méthode d'optimisation élimine les branches inutiles de l'arbre de recherche, améliorant ainsi l'efficacité de la recherche de solutions optimales.
* La programmation dynamique : Dans le cas de grilles de taille fixe et réduite, la programmation dynamique peut être utilisée pour trouver des solutions optimales.
* Les méthodes Monte Carlo : Pour des grilles de grande taille, des méthodes Monte Carlo peuvent être employées pour générer des dispositions aléatoires et vérifier leur conformité aux exigences.
* Les heuristiques et les métaheuristiques : Des approches telles que le Recuit Simulé, les Algorithmes Génétiques et la Recherche Tabou peuvent être appliquées pour rechercher des solutions approximatives ou améliorer les solutions existantes.

Chaque problème spécifique dans les Pavages de Polyominos peut nécessiter l'application d'un algorithme ou d'une méthode différente en fonction de sa portée et de ses exigences particulières. Ce mélange d'approches permettra d'explorer efficacement et de résoudre les défis posés par le projet.

## **Conception de l'interface utilisateur**

### **Version 2D**

La conception de l'interface utilisateur pour la partie 2D de notre projet se veut intuitive et fonctionnelle, permettant une interaction aisée avec les polyominos. L'interface sera structurée comme suit :

* Barre Latérale Gauche (Sidebar Gauche):
  + Occupant environ 10% de la largeur totale de l'écran, cette barre latérale sera dédiée aux réglages et ajustements de la grille.
  + Elle permettra à l'utilisateur de modifier la taille de la grille sur laquelle les polyominos seront placés.
  + Des options supplémentaires pourront y être intégrées, telles que la sélection des couleurs ou d'autres paramètres personnalisables.
* Barre Latérale Droite (Sidebar Droite):
  + Cette barre, occupant 15% de la largeur de l'écran, servira de zone de création et de stockage des polyominos.
  + L'utilisateur pourra y sélectionner, créer et modifier les formes des polyominos avant de les placer sur la grille.
  + Elle comprendra également des fonctionnalités pour sauvegarder ou charger des configurations de polyominos.
* Zone Centrale - Grille de Placement:
  + La partie restante de l'écran sera dédiée à la grille principale où les polyominos seront placés.
  + Cette grille interactive permettra un placement facile et précis des pièces.
  + Elle affichera en temps réel les modifications apportées par l'utilisateur, offrant une visualisation claire du pavage en cours de création.

L'interface sera conçue pour être responsive, s'adaptant aux différents formats d'écran et offrant une expérience utilisateur optimale sur les appareils de bureau et mobiles. L'accent sera mis sur la simplicité et l'efficacité, afin de rendre l'outil accessible à un large éventail d'utilisateurs, des passionnés de géométrie aux étudiants et enseignants.

La combinaison de ces éléments vise à offrir une expérience utilisateur enrichissante et éducative, facilitant l'exploration et la compréhension des concepts de pavages de polyominos.

## **Identification des structures de données**

* Structures pour Polyominos:
  + Objets Polyomino: Chaque polyomino sera représenté par un objet contenant des informations telles que sa forme, sa taille, sa couleur, et sa position sur la grille.
  + Matrice de Configuration: Une matrice (ou tableau 2D) sera utilisée pour représenter la configuration spatiale de chaque polyomino, facilitant ainsi le calcul de leur placement sur la grille.
* Structures pour la Grille:
  + Grille de Pavage: La grille sera représentée par une matrice, où chaque cellule indiquera l'état d'occupation (vide ou occupé par un polyomino).
  + Gestion des États: Un système de gestion des états permettra de suivre les modifications apportées à la grille, y compris l'ajout ou la suppression de polyominos.
* Gestion des Données Utilisateur:
  + Des structures telles que des listes ou des files d'attente seront utilisées pour stocker les configurations de polyominos créées ou modifiées par les utilisateurs.
  + Un système de sauvegarde et de chargement permettra de stocker et de récupérer ces configurations.

Ces structures de données doivent être conçues pour être flexibles, efficaces et extensibles, afin de supporter les fonctionnalités prévues et les éventuelles évolutions du projet.

## **Évaluer la fonctionnalité et la faisabilité de la conception**

L'évaluation de la fonctionnalité et de la faisabilité de la conception de notre projet Pavages de Polyominos est cruciale pour assurer le succès et la pertinence de l'outil.

* Analyse de la Fonctionnalité:
  + Adéquation avec les Objectifs: Vérifier si les fonctionnalités planifiées correspondent aux objectifs pédagogiques et de recherche du projet.
  + Utilité Utilisateur: Évaluer si l'interface et les fonctionnalités répondent aux besoins et aux attentes des utilisateurs cibles, tels que les étudiants, les enseignants et les passionnés de géométrie.
* Analyse de la Faisabilité Technique:
  + Complexité Technique: Examiner la complexité des structures de données et des algorithmes nécessaires pour la mise en œuvre des fonctionnalités.
  + Ressources Requises: Estimer les ressources nécessaires en termes de développement, y compris le temps, les compétences techniques

et le matériel requis. Il est crucial de s'assurer que ces ressources sont disponibles et suffisantes pour mener à bien le projet.

# **VI. Développement (Sera mis à jour à l'avenir)**

## **Déployer le code source et créer des fonctions selon les besoins**

Code

## **Vérifier et corriger les erreurs**

Code

## **Assurer le respect des normes de programmation et des règles d'exécution des projets**

Code

## **Signaler les progrès et les problèmes rencontrés**

Code

# **VII. Tests et validation (Sera mis à jour à l'avenir)**

* Effectuer des tests de qualité, des tests d'intégration et des tests système
* Tester l'exactitude et les performances de l'application
* Assurer la compatibilité multiplateforme (si nécessaire)
* Confirmer que le projet répond aux exigences et objectifs énoncés

# **VIII. Documentation (Sera mis à jour à l'avenir)**

* Créer la documentation de l'application
* Rédiger la documentation technique sur le code source et la structure du projet
* Créer des documents de rapport de projet

# **IX. Formation (Sera mis à jour à l'avenir)**

* Formation des utilisateurs finaux (si nécessaire)
* Formation sur la gestion et la maintenance de l'application

# **X. Livraison (Sera mis à jour à l'avenir)**

Présenter la demande dûment remplie au client ou au promoteur

Remettre les documents et les actifs du projet

Vérifier les exigences du client

# **XI. Maintenance et assistance (Sera mis à jour à l'avenir)**

Déterminer les plans de maintenance et de support post-déploiement

Énoncez clairement votre engagement envers le support client

# **XII. Conclusion**

Lors de cet état de l'art, nous avons acquis de nombreuses connaissances intéressantes sur les polyominos et les polycubes en explorant leurs propriétés, classifications et implications dans le domaine du pavage. Nous avons constaté comment, malgré leur simplicité, les polyominos en deux dimensions ont pu engendrer des problèmes mathématiques.

L'introduction des polycubes en trois dimensions a considérablement élargi le champ des possibles, créant des défis algorithmiques encore plus complexes, notamment grâce à la connectivité entre les cubes dans l'espace tridimensionnel, ouvrant ainsi de nouvelles perspectives et permettant des constructions plus élaborées et stimulantes. En comprenant les différentes classifications de ces structures, qu'elles soient "libres", "fixes" ou "unilatérales", nous avons saisi la richesse de ces formes.

Notre document s'est également penché sur divers outils et bibliothèques disponibles, tels que Three.js et l'API Canvas en JavaScript, afin d'explorer des solutions potentielles dans le contexte du projet tuteuré. Ces outils nous permettront de représenter visuellement les polyominos et les polycubes.

Nous avons également consacré du temps à aborder les différentes notions liées aux pavages en 2D et 3D, et nous avons réussi à envisager et à proposer des pistes de réflexion pour l'intégration future du programme dans la prochaine partie du projet tuteuré.

En résumé, cet état de l'art approfondi nous a préparés à aborder notre projet tuteuré avec une compréhension étendue des enjeux, des défis algorithmiques et des possibilités offertes par les polyominos et les polycubes. C'est avec ces connaissances que nous pourrons entreprendre la prochaine étape de ce projet, à savoir l'intégration du programme.

# **XIII. Annexes (Sera mis à jour à l'avenir)**

Documents, diagrammes, code source ou autres informations liées au projet

# **XIV. Bibliographie**

### **Sources utilisé pour les images et les notes**

* Farès Belhadj. *Modèle pour projet tuteuré ou* [*rapport de stage*](https://expreg.org/amsi/C/LIV/dl/memoire.pdf).

### **Sources intéressantes dans le réalisation de notre document**

* Kevin Gong.[*Applying Parallel Programming to the Polyomino Problem*](http://kevingong.com/Polyominoes/ParallelPoly.html)*.*
* Joseph Myers.[*Polyform Tiling*](https://www.polyomino.org.uk/mathematics/polyform-tiling/)*.*
* Alexandre Blondin Massé , Amadou Makhtar Tall, Hugo Tremblay. [*On the Arithmetics of Discrete Figures*](https://constellation.uqac.ca/id/eprint/6126/1/lata2014-prime.pdf).
* Olivier Boudini. [*Z-tiling of polyominoes and standard basis*](https://igm.univ-mlv.fr/~fpsac/FPSAC02/ARTICLES/Bodini.pdf)*.*
* *Article* [*Wikipedia : Polyomino*](https://en.wikipedia.org/wiki/Polyomino).
* *Article* [*Wikipedia : JavaScript*](https://fr.wikipedia.org/wiki/JavaScript).
* *Page officielle de la librairie* [*Three.js*](https://threejs.org/)*.*

1. : Une fusion entre les échecs et les dames voir les règles ici : [How to Play Chesskers](https://chesskers.lol/rules) [↑](#footnote-ref-0)
2. : les informations viennent de cette page wikipedia [Polyomino — Wikipédia](https://fr.wikipedia.org/wiki/Polyomino) ainsi que cette page : [polyominos,introduction aux dominos, pentominos et autres](http://villemin.gerard.free.fr/Puzzle/minoPoly.htm) et [Polyform tiling (polyomino.org.uk)](https://www.polyomino.org.uk/mathematics/polyform-tiling/) [↑](#footnote-ref-1)
3. : Solomon W. Golomb créateur du terme “polyominos” présente certaines impossibilité lié au plateau du jeu de dame dans son livre : [Polyominoes – Solomon W.Golomb](about:blank) [↑](#footnote-ref-2)
4. : les informations viennent de cette page [wikipedia Polycube](https://en.wikipedia.org/wiki/Polycube) [↑](#footnote-ref-3)