



# v1.0 **Projet CY-Crush saga**

CLASSE PRE-ING1 • 2022-2023

AUTEUR Eva ANSERMIN

E-MAIL [eva.ansermin@cyu.fr](mailto:eva.ansermin@cyu.fr)

## DESCRIPTION

Réaliser un programme C vous permettant de jouer à un jeu de type « match 3 » sur un terminal.

Nous vous proposons plusieurs versions de ce projet qui ont des règles différentes. Si vous choisissez ce projet, la version vous sera imposée.

Ce projet implique une bonne gestion des tableau 2D.

En plus de respecter son propre cahier des charges, ce projet devra également respecter les consignes générales données pour tous les projets.

## CAHIER DES CHARGES MINIMAL

- Un exemple d'interface du jeu :

```

      A B C D E F G H I J
1 | U X I O U X O U I O
2 | X I X X U X I X O X
3 | I O I O X O I O X I
4 | I X O I O U X O I O
5 | O O I X X I I U U U
6 | O X U X I X X O X I
7 | U I U I U U I U U O
8 | O X O U I X X O I O
9 | U X I U X O O U O X
10| X U O O I U O O U O

Quelle case voulez vous déplacer?

```

- En début de partie, le programme va créer une grille MxN contenant entre 4 et 6 symboles différents. Cette grille devra être générée de manière aléatoire tout en s'assurant que **jamais 3 symboles identiques soient côte à côte**.
- La taille de la grille et le nombre de symboles différents sont définis par l'utilisateur avant de commencer à jouer.
- À chaque tour, le joueur devra choisir deux cases à échanger afin de former une ligne de 3 symboles identiques (les règles de déplacement varient selon la version).
- Lorsque 3 symboles identiques (ou plus) sont alignés horizontalement ou verticalement, ils doivent disparaître. La gravité va faire se déplacer les autres symboles restants. Cela peut entraîner une réaction en chaîne qui devra pouvoir être gérée par le programme.

- Le programme devra détruire tous les symboles avant d'appliquer la gravité.
- Le remplissage de la grille en cours de partie dépend de la version.
- Un déplacement qui n'aligne pas au minimum 3 symboles identiques n'est pas autorisé par le programme.
- Le joueur doit marquer des points à chaque fois qu'il supprime des symboles.
- La partie se termine lorsqu'il n'est plus possible d'effectuer un déplacement (voir les différentes versions). Le programme devra vérifier qu'il n'y a plus de possibilité de déplacements.
- HIGH SCORES :
  - À la fin d'une partie, en cas de victoire, le nom du joueur et son temps de jeu doivent être stockés dans un fichier.
  - À la fin de chaque partie, les meilleurs scores enregistrés doivent être affichés.
- SAUVEGARDE :
  - À tout moment, le joueur pourra sauvegarder la partie. L'état de sa grille ainsi que son score courant seront alors stockés dans un fichier et la partie pourra être rechargée plus tard.
  - Vous pouvez décider de ne pouvoir sauvegarder qu'une partie à la fois ou bien autant que souhaité. Auquel cas le joueur devra indiquer le nom ou le numéro de son fichier sauvegarde.

## VERSIONS

- Chaque groupe qui aura choisi ce projet se verra assigner une version pour chacun des thèmes ci-dessous (Déplacement, Suppression des symboles, comportement de la grille, ...). Tous les groupes auront néanmoins à implémenter les critères communs.
- **DEPLACEMENTS :**
  - Version D1 : un symbole peut être échangé uniquement avec une case voisine (sur une ligne horizontale ou verticale).
  - Version D2 : un symbole peut être échangé avec n'importe quelle case du plateau.
- **SUPPRESSION DES SYMBOLES :**
  - Version S1 : En plus des alignements horizontaux et verticaux classiques, vous devrez supprimer les symboles alignés en diagonale (sans changer le mode de déplacement précédent).
  - Version S2 : En plus des alignements horizontaux et verticaux classiques à l'intérieur de la zone de jeu, vous pouvez détruire des symboles qui sont de part et d'autre de la zone de jeu : exemple, si vous avez 2 symboles rouges alignés horizontalement sur le bord gauche du plateau ET vous avez un troisième symbole de même couleur sur le bord droit du plateau sur la même ligne, ces 3 symboles sont détruits car considérés comme connectés. Ici il faut considérer que les bords verticaux sont côte à côte, et que les bords haut et bas sont l'un en dessous de l'autre.
- **AJOUT DE NOUVEAUX SYMBOLES :**
  - Version A1 : la grille doit toujours être remplie : lorsque des symboles sont supprimés, des nouveaux symboles sont générés de manière aléatoire. Le but du jeu est de faire le plus grand nombre de coups possible sans être bloqué.
  - Version A2 : Les symboles supprimés ne sont pas remplacés. Le but du jeu est de finir avec une grille ne comportant que deux symboles ou moins en ayant fait le moins de coups possible.

- **GESTION DE LA GRAVITE :**
    - Version G1 : La gravité est dirigée vers le bas.
    - Version G2 : la gravité est dirigée vers la gauche.
    - Version G3 : la gravité est dirigée vers le haut.
    - Version G4 : La gravité est dirigée vers la droite.
- 

## AMELIORATIONS POSSIBLES

- Ajouter des couleurs dans l'affichage de la grille pour mieux distinguer les différents symboles.
  - Implémenter d'autre forme de grille. Voir pouvoir dessiner la forme de grille souhaitée dans un fichier.
  - Partie locale à deux joueurs : deux joueurs jouent sur la même grille : celui qui arrive à bloquer l'autre (ne plus lui laisser de déplacement possible), a gagné la partie.
  - Ajout de joker pour permettre des échanges de cases gratuitement (sans que cela n'implique l'alignement de 3 symboles identiques). Ces jokers pourront être donnés en nombre limité en début de partie et/ou être gagnés au cours de celle-ci en fonction de règles à définir.
  - Vous pouvez implémenter d'autres améliorations de votre choix.
- 

## RESSOURCES UTILES

### GitHub

- <https://github.com>
- <https://docs.github.com/en/get-started/quickstart/hello-world>

### Modifier les couleurs du terminal Linux

- [https://en.wikipedia.org/wiki/ANSI\\_escape\\_code#Colors](https://en.wikipedia.org/wiki/ANSI_escape_code#Colors)
  - <http://sdz.tdct.org/sdz/des-couleurs-dans-la-console-linux.html>
-