

RAPPORT PROJET CDAA

Yann TROU | TD2 Groupe B

Octobre 2020

Table des matières

I | Cahier des charges

1. Fonctions

2. Structures

II | Analyse fonctionnelle

1. Classes et packages

2. Affichage et CLI

3. Interactions et commandes

III | Manuel d'utilisation

1. Commandes

2. Dépannage

IV | Panier d'objets génériques

I | Cahier des charges

1. Fonctions

On peut définir le cahier des charges à partir des sujets de TP et TD en plus de l'énoncé du sujet.

L'objectif est pour le moment de créer un prototype de site de vente de jeux vidéos, consoles, et autres. Ce prototype se limitera à la ligne de commande pour l'interface et de la gestion de catalogue. Le sujet indique que l'on doit ajouter un autre type d'objets gérés en plus des jeux. J'ai choisi des Consoles. Ainsi, les fonctionnalités attendues sont :

- Ajouter un jeu / une console
- Supprimer un jeu / une console
- Visualiser tous les jeux / toutes les consoles
- Rechercher dans les jeux ou les consoles avec des paramètres spécifiques
- Trier les jeux ou les consoles par champ
- Pouvoir gérer des jeux rétro

2. Structures

Les structures de données proposées par le tp sont les suivantes : un catalogue regroupant les ensembles d'objets vendables, ces mêmes ensembles utilisant `List<>` pour stocker leurs éléments. Ensemble est une classe abstraite regroupant des méthodes pour ajouter, supprimer, trier et rechercher. Les implémentations de cette classe sont `EnsembleJeux` et `EnsembleConsoles` respectivement pour les jeux et les consoles.

II | Analyse Fonctionnelle

Faute de pouvoir le faire tenir dans ce fichier pdf, le graphique détaillé des classes, packages et relations se trouve en annexe dans le fichier `DiagrammeClasses.jpg`.

1. Classes et packages

A partir du cahier des charges, nous pouvons commencer à définir les grandes lignes de notre programme. Nous aurons besoin d'un ensemble de classes représentant les jeux et les consoles, un package contenant les structures de données, un package pour l'application des commandes.

Pour les jeux nous aurons donc :

- une classe `Jeu` pour représenter un jeu
- une classe `JeuRetro` héritant de `Jeu` pour représenter un jeu rétro
- une énumération `Genre` pour indiquer le genre de jeu
- une classe `EnsembleJeu` qui hérite de `Ensemble` et se spécialise pour gérer des jeux.

Pour les consoles nous aurons un jeu de classes similaires

Nous aurons une classe `Catalogue` qui contiendra un ensemble de chaque produit et servira de point d'accès vers ceux-ci

Pour pouvoir trier ces objets, nous allons utiliser un tri quicksort dans la classe du même nom.

Afin de pouvoir spécifier l'attribut sur lequel trier les objets, on va créer une interface `IFieldComparable` avec une seule méthode, `CompareFieldTo`. l'implémentation de cette méthode sera faite dans les classes `Jeu` et `Console`.

Pour pouvoir faire des ensembles d'objets, il nous faut la classe abstraite `ensemble`

Il nous faut aussi une classe permettant de faire interagir toutes ces autres classes, ce sera le rôle de la classe `Program`.

Enfin, les classes `CommandesConsoles` et `CommandesJeux` serviront à séparer les implémentations des commandes de la classe `Program` pour l'alléger.

Nous pouvons en déduire les packages suivants :

- `CLI` : classes liées à l'affichage dans la console
- `commands` : classes pour traiter les commandes
- `Core` : noyau du programme
 - `consoles` : classes liées à la représentation d'une console
 - `jeux` : classes liées à la représentation d'un jeu
- `datastructures` : structures de données génériques utilisées dans d'autres parties du programme
- `doc` : fichiers texte ayant servi à l'élaboration du programme

2. Affichage et CLI

Pour plus de confort et de lisibilité, j'ai eu l'idée d'implémenter un système d'affichage dans la console se contrôlant au clavier et reprenant des éléments similaires à ceux que l'on peut trouver dans de vrais programmes avec une interface.

Maintenant, pour le traitement et l'affichage, je me suis inspiré de [cette vidéo \(cliquez\)](#). Les principaux éléments que j'ai retenus sont l'utilisation d'une pile pour stocker et afficher avec de la profondeur les éléments d'interface et aussi le fait que les éléments se dessinent eux-mêmes. L'utilisation d'une

pile permet de ne donner le contrôle qu'à un seul élément, celui tout en haut de la pile donc au premier plan.

La différence principale tient au fait que là où il a un moteur pour afficher des images en couleur, je n'ai que la console. Mais heureusement, la console possède une méthode `SetCursorPosition`, permettant de placer le curseur n'importe où dans les limites de la fenêtre et d'écrire ce que l'on veut. Nous avons les bases du système de CLI.

J'ai pour commencer créé la classe `CLIManager`, qui contient et gère la pile de `CLIElement`, classe mère de tous les éléments. `CLIManager` ne contient pas grand chose, des méthodes pour ajouter et supprimer des `CLIElement`, une méthode pour mettre à jour l'affichage et une méthode `Loop`. Cette méthode est importante car c'est la boucle principale du programme : mettre à jour le contenu de la table, mettre à jour l'affichage, récupérer la touche que l'utilisateur a pressé et la transmettre à l'élément en haut du tas, et recommencer.

Maintenant, parlons des éléments d'interface disponibles et de leurs classes.

Les éléments qui ont été implémentés sont

- une fenêtre vide : classe `CLIWindow`. au final il ne s'agit que d'une bordure
- une fenêtre modale : classe `CLIInputWindow`. permet à l'utilisateur de rentrer du texte, et de le récupérer dans le programme
- un menu (vu comme une liste) : classe `CLIMenu`. permet de parcourir des commandes et d'en sélectionner une.
- un tableau : classe `CLITable`. permet d'afficher sous forme de table des données.

L'un des problèmes que j'ai rencontré est la vitesse d'affichage. En effet, à un point on voyait tous les éléments se dessiner plutôt que l'affichage soit fait instantanément. Une première solution a été de concaténer des chaînes de caractères en une seule pour réduire le nombre d'appels à `SetCursorPosition` et `Write` mais aussi de supprimer les boucles d'affichage horizontal. Mais cette solution a été remplacée par l'utilisation de `StringBuilder`. Car, après un tour dans la documentation microsoft (ici précisément) j'ai appris que les strings étaient immutables et qu'à chaque opération, ils étaient copiés et l'opération était appliquée sur la copie. De plus, le fait que la méthode `Append()` puisse ajouter *n* fois un caractère permet d'optimiser encore mieux le code. Avec tout cela, nous avons retrouvé des vitesses d'affichage quasiment instantanées.

Un des problèmes récurrent et que vous pouvez rencontrer, tient du fait que la taille de la console varie en fonction de chaque écran et de chaque résolution et des erreurs dues à des débordement peuvent apparaître. Dans certains cas, la console n'est pas du tout supportée (comme sur mon mac). Si le programme ne fonctionne pas, s'il vous plaît jetez un coup d'oeil au manuel ou au pire contactez moi.

3. Interactions et commandes

Ici, nous allons détailler comment les commandes sont traitées.

Déjà, chaque commande est identifiée par un entier. Elles ont été définies en amont du codage et sont réparties ainsi :

- de -infini à -1 : commandes générales
- -1 : suppression de la recherche et affichage de toute la table des jeux

- -2 : suppression de la recherche et affichage de toute la table des consoles.
- de 0 à 99 : commandes pour les jeux
 - 0 : ajouter un jeu
 - 1 : supprimer un jeu
 - 2 : trier les jeux
 - 3 : faire une recherche dans les jeux
 - de 20 à 27 : chaque commande correspond à un champ sur lequel appliquer le tri
 - de 30 à 37 : chaque commande correspond à un champ sur lequel appliquer une recherche
- de 100 à 199 : commandes pour les consoles
 - 100 : ajouter une console
 - 101 : supprimer une console
 - 102 : trier les consoles
 - 103 : rechercher une console
 - de 120 à 127 : chaque commande correspond à un champ sur lequel appliquer le tri
 - de 130 à 137 : chaque commande correspond à un champ sur lequel appliquer une recherche

Le cycle de vie d'une commande fonctionne ainsi : dans un CLIMenu, l'utilisateur appuie sur entrée. La commande associée à la position du curseur est envoyée à Program via la méthode statique `dispatchCommand`. Ensuite, en fonction de l'état du programme, soit la commande est traitée dans `commandesJeux` si l'état du programme est à 0, soit la commande est traitée dans `commandesJeux` si l'état du programme est à 1. Dans n'importe quel état elle est testée par rapport aux commandes générales.

Les commandes s'appliquent de manière similaires aux jeux et aux consoles.

Pour ajouter un jeu, on va créer une `CLIInputWindow`, l'initialiser pour qu'elle ai le message correspondant et la placer dans la pile d'affichage. Ensuite, on met à jour l'affichage, on place l'entrée clavier sur celle ci. Une fois que l'utilisateur a rentré du texte, on le récupère, on le cast si nécessaire et on le set dans le jeu. On recommence pour chaque attribut. Une fois qu'ils sont tous rentrés, on supprime la `CLIInputWindow` de la pile d'affichage et on ajoute le jeu dans le catalogue.

III | Manuel

Pour retrouver toutes les images dans leur taille originale, allez voir /doc/Rapport/ dans le dossier du code source du projet.



FIGURE 1 – Lorsque le programme démarre, il vous demande quel programme exécuter

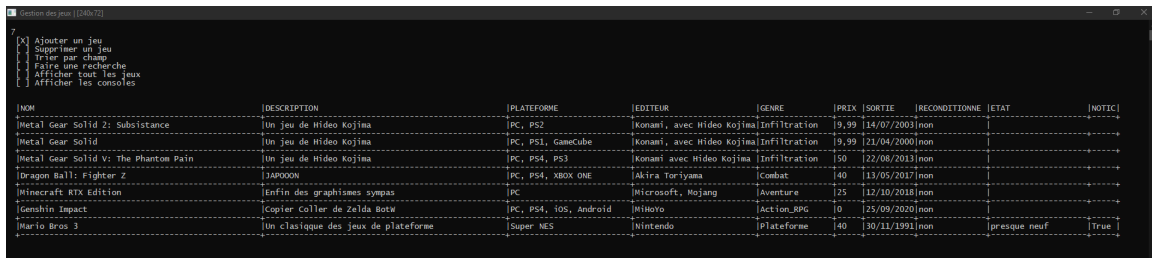


FIGURE 2 – On peut voir le menu pour la sélection des commandes ainsi que la table qui affiche tous les jeux. Pour se déplacer dans le menu, on utilise les flèches haut et bas et entrée pour sélectionner une commande

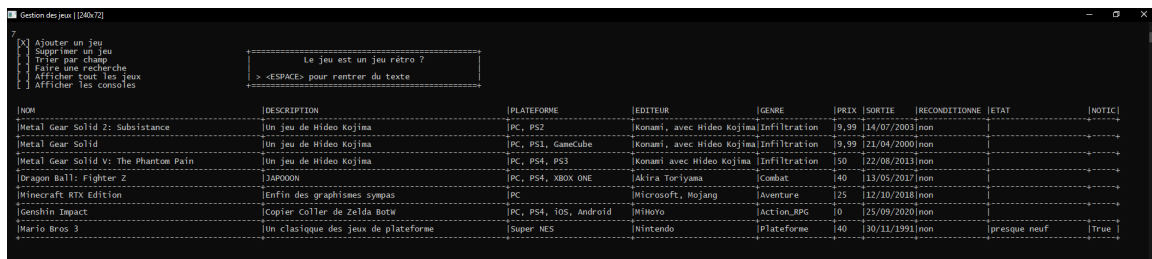
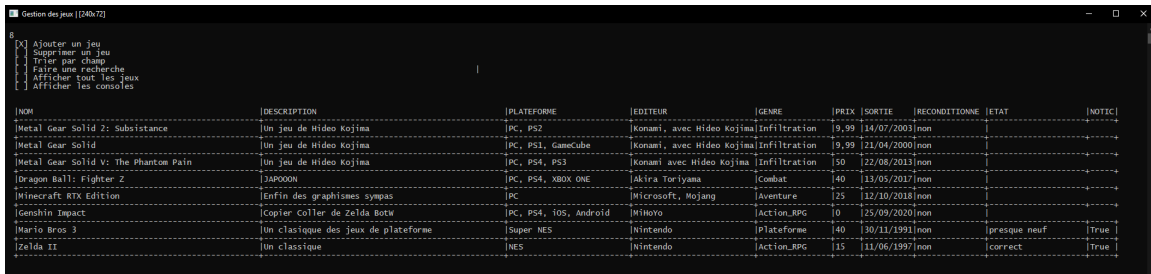


FIGURE 3 – Lors de l'ajout d'un jeu, une CLIInputWindow s'affiche pour indiquer quel paramètre entrer et dans quel format

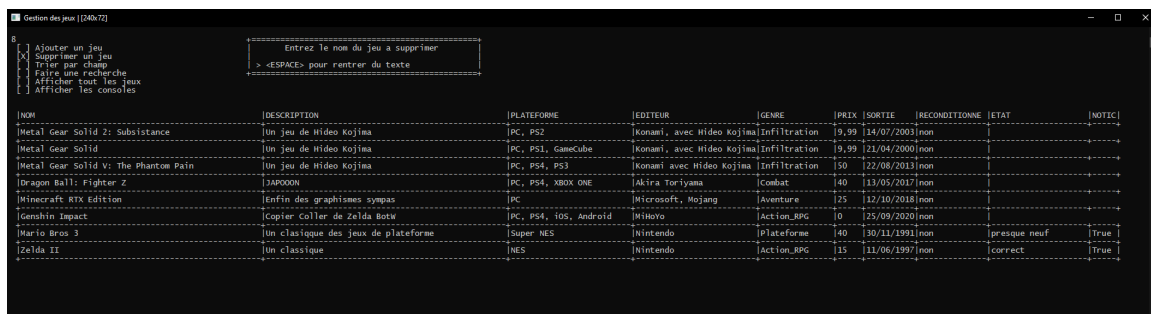


8

- Ajouter un jeu
- Supprimer un jeu
- Trier par champ
- Faire une recherche
- Afficher tout les jeux
- Afficher les consoles

NOM	DESCRIPTION	PLATEFORME	EDITEUR	GENRE	PREIX	SORTIE	RECONDITIONNE	ETAT	NOTIC
Metal Gear Solid 2: Subsistence	Un jeu de Hideo Kojima	PC, PS2	Konami, avec Hideo Kojima	Infiltration	9,99	14/07/2003	non		
Metal Gear Solid	Un jeu de Hideo Kojima	PC, PS1, GameCube	Konami, avec Hideo Kojima	Infiltration	9,99	21/04/2000	non		
Metal Gear Solid V: The Phantom Pain	Un jeu de Hideo Kojima	PC, PS4, PS3	Konami avec Hideo Kojima	Infiltration	50	22/08/2013	non		
Dragon Ball: Fighter Z	JAPPOON	PC, PS4, XBOX ONE	Akira Toriyama	Combat	40	11/05/2017	non		
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		
Genshin Impact	Copier Collier de Zelda BotW	PC, PS4, IOS, Android	MiHoYo	Action_RPG	0	25/09/2020	non		
Mario Bros 3	Un classique des jeux de plateforme	Super NES	Nintendo	Plateforme	40	30/11/1991	non	presque neuf	True
Zelda II	Un classique	NES	Nintendo	Action_RPG	15	11/06/1997	non	correct	True

FIGURE 4 – Une fois tous les paramètres rentrés, le jeu est crée et est ajouté dans la table qui sera actualisée. Ici, j’ai ajouté Zelda II.

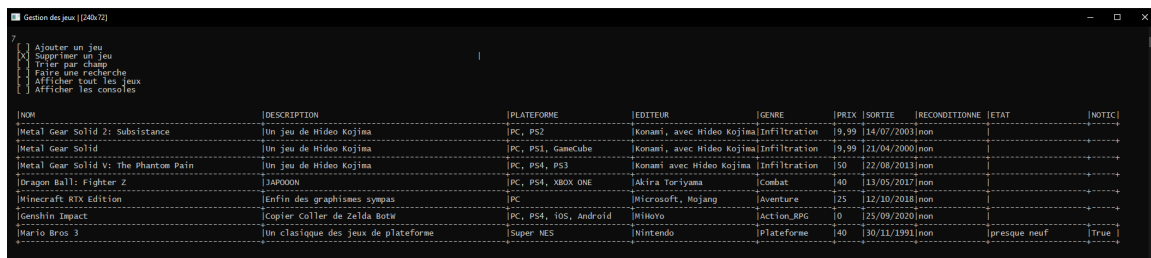


6

Entrez le nom du jeu à supprimer
> <SPACE> pour rentrer du texte

NOM	DESCRIPTION	PLATEFORME	EDITEUR	GENRE	PREIX	SORTIE	RECONDITIONNE	ETAT	NOTIC
Metal Gear Solid 2: Subsistence	Un jeu de Hideo Kojima	PC, PS2	Konami, avec Hideo Kojima	Infiltration	9,99	14/07/2003	non		
Metal Gear Solid	Un jeu de Hideo Kojima	PC, PS1, GameCube	Konami, avec Hideo Kojima	Infiltration	9,99	21/04/2000	non		
Metal Gear Solid V: The Phantom Pain	Un jeu de Hideo Kojima	PC, PS4, PS3	Konami avec Hideo Kojima	Infiltration	50	22/08/2013	non		
Dragon Ball: Fighter Z	JAPPOON	PC, PS4, XBOX ONE	Akira Toriyama	Combat	40	11/05/2017	non		
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		
Genshin Impact	Copier Collier de Zelda BotW	PC, PS4, IOS, Android	MiHoYo	Action_RPG	0	25/09/2020	non		
Mario Bros 3	Un classique des jeux de plateforme	Super NES	Nintendo	Plateforme	40	30/11/1991	non	presque neuf	True
Zelda II	Un classique	NES	Nintendo	Action_RPG	15	11/06/1997	non	correct	True

FIGURE 5 – Quand on veut supprimer un jeu, une CLIInputWindow s’affiche et demande le nom du jeu. Ce sera le seul paramètre demandé.

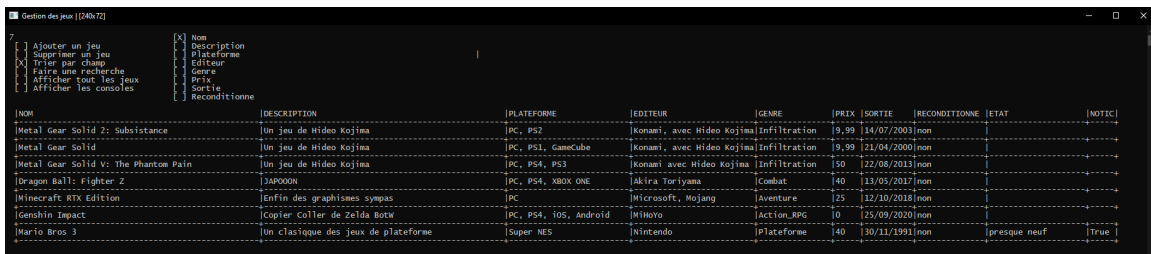


7

- Ajouter un jeu
- Supprimer un jeu
- Trier par champ
- Faire une recherche
- Afficher tout les jeux
- Afficher les consoles

NOM	DESCRIPTION	PLATEFORME	EDITEUR	GENRE	PREIX	SORTIE	RECONDITIONNE	ETAT	NOTIC
Metal Gear Solid 2: Subsistence	Un jeu de Hideo Kojima	PC, PS2	Konami, avec Hideo Kojima	Infiltration	9,99	14/07/2003	non		
Metal Gear Solid	Un jeu de Hideo Kojima	PC, PS1, GameCube	Konami, avec Hideo Kojima	Infiltration	9,99	21/04/2000	non		
Metal Gear Solid V: The Phantom Pain	Un jeu de Hideo Kojima	PC, PS4, PS3	Konami avec Hideo Kojima	Infiltration	50	22/08/2013	non		
Dragon Ball: Fighter Z	JAPPOON	PC, PS4, XBOX ONE	Akira Toriyama	Combat	40	11/05/2017	non		
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		
Genshin Impact	Copier Collier de Zelda BotW	PC, PS4, IOS, Android	MiHoYo	Action_RPG	0	25/09/2020	non		
Mario Bros 3	Un classique des jeux de plateforme	Super NES	Nintendo	Plateforme	40	30/11/1991	non	presque neuf	True

FIGURE 6 – Une fois le nom saisi, le jeu est supprimé et la table est mise à jour.



7

- Ajouter un jeu
- Supprimer un jeu
- Trier par champ
- Faire une recherche
- Afficher tout les jeux
- Afficher les consoles

- Nom
- Description
- Plateforme
- Editeur
- Genre
- Prix
- Sortie
- Reconditionne

NOM	DESCRIPTION	PLATEFORME	EDITEUR	GENRE	PREIX	SORTIE	RECONDITIONNE	ETAT	NOTIC
Metal Gear Solid 2: Subsistence	Un jeu de Hideo Kojima	PC, PS2	Konami, avec Hideo Kojima	Infiltration	9,99	14/07/2003	non		
Metal Gear Solid	Un jeu de Hideo Kojima	PC, PS1, GameCube	Konami, avec Hideo Kojima	Infiltration	9,99	21/04/2000	non		
Metal Gear Solid V: The Phantom Pain	Un jeu de Hideo Kojima	PC, PS4, PS3	Konami avec Hideo Kojima	Infiltration	50	22/08/2013	non		
Dragon Ball: Fighter Z	JAPPOON	PC, PS4, XBOX ONE	Akira Toriyama	Combat	40	11/05/2017	non		
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		
Genshin Impact	Copier Collier de Zelda BotW	PC, PS4, IOS, Android	MiHoYo	Action_RPG	0	25/09/2020	non		
Mario Bros 3	Un classique des jeux de plateforme	Super NES	Nintendo	Plateforme	40	30/11/1991	non	presque neuf	True

FIGURE 7 – Pour trier tous les jeux, il faut sélectionner la commande, puis sélectionner le champ sur lequel trier.

NOM	DESCRIPTION	PLATEFORME	EDEITEUR	GENRE	PRIX	SORTIE	RECONDITIONNEMENT	ETAT	NOTICE
Genshin Impact	Copier Coller de Zelda Botw	PC, PS4, iOS, Android	Mihoyo	Action_RPG	0	25/09/2020	non		
Metal Gear Solid	Un jeu de Hideo Kojima	PC, PS1, GameCube	Konami, avec Hideo Kojima	Infiltration	9,99	21/04/2000	non		
Metal Gear Solid 2: Subsistence	Un jeu de Hideo Kojima	PC, PS2	Konami, avec Hideo Kojima	Infiltration	9,99	14/07/2003	non		
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		
Mario Bros 3	Un classique des jeux de plateforme	Super NES	Nintendo	Plateforme	40	30/11/1991	non	presque neuf	True
Dragon Ball: Fighter Z	JAPPOON	PC, PS4, XBOX ONE	Akira Toriyama	Combat	40	13/05/2017	non		
Metal Gear Solid V: The Phantom Pain	Un jeu de Hideo Kojima	PC, PS4, PS3	Konami avec Hideo Kojima	Infiltration	50	22/08/2013	non		

FIGURE 8 – Ici, j’ai choisi de trier les jeux par prix.

NOM	DESCRIPTION	PLATEFORME	EDEITEUR	GENRE	PRIX	SORTIE	RECONDITIONNEMENT	ETAT	NOTICE
Genshin Impact	Copier Coller de Zelda Botw	PC, PS4, iOS, Android	Mihoyo	Action_RPG	0	25/09/2020	non		
Metal Gear Solid	Un jeu de Hideo Kojima	PC, PS1, GameCube	Konami, avec Hideo Kojima	Infiltration	9,99	21/04/2000	non		
Metal Gear Solid 2: Subsistence	Un jeu de Hideo Kojima	PC, PS2	Konami, avec Hideo Kojima	Infiltration	9,99	14/07/2003	non		
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		
Mario Bros 3	Un classique des jeux de plateforme	Super NES	Nintendo	Plateforme	40	30/11/1991	non	presque neuf	True
Dragon Ball: Fighter Z	JAPPOON	PC, PS4, XBOX ONE	Akira Toriyama	Combat	40	13/05/2017	non		
Metal Gear Solid V: The Phantom Pain	Un jeu de Hideo Kojima	PC, PS4, PS3	Konami avec Hideo Kojima	Infiltration	50	22/08/2013	non		

FIGURE 9 – La recherche fonctionne comme le tri. Sélectionnez la commande, le champ puis une CLIInputWindow demande la valeur de ce que vous voulez rechercher.

NOM	DESCRIPTION	PLATEFORME	EDEITEUR	GENRE	PRIX	SORTIE	RECONDITIONNEMENT	ETAT	NOTICE
Minecraft RTX Edition	Enfin des graphismes sympas	PC	Microsoft, Mojang	Aventure	25	12/10/2018	non		

FIGURE 10 – Le résultat d’une recherche s’affiche à la place de la table. Ici j’ai recherché les jeux n’ayant que PC comme plateforme. Pour pouvoir exécuter une autre commande, vous devez sélectionner la commande afficher tous les jeux.

NOM	FABRIQUANT	GENERATION	SORTIE	PORTS	SUPPORT	TYPE
Switch	Nintendo	8	01/03/2017	8	Cartouche	Hybride

FIGURE 11 – Lorsque l’on choisit la commande afficher les consoles, le gestionnaire affiche les commandes des consoles ainsi qu’une CLITable affichant tous les jeux présents dans le catalogue. Les commandes se comportent exactement comme celles vues précédemment donc nous n’allons pas en parler.

2. Dépannage

Il peut arriver que le programme plante. Comme indiqué au dessus, une des erreurs que vous risquez le plus de rencontrer est liée à la console. Mais il y en a quelques autres qui peuvent être rencontrées. Notamment lors de la saisie des attributs pendant l'ajout, si le format n'est pas celui prévu, le programme plantera.

Une corruption de l'affichage peut apparaître : du texte là il n'est sensé rien y avoir. Son origine est inconnue, si elle est constatée, effectuez une ou deux commandes pour qu'elle disparaisse.

Ce sont les erreurs que nous avons rencontrées et qui sont encore présentes, dans l'attente d'un correctif.

IV | Panier générique d'objets

Pour terminer, nous allons parler du panier d'objets. Il utilise un type générique qui doit implémenter `IComparable` et `IEquatable` pour pouvoir trier le tableau et vérifier l'existence de doublons. La taille maximum du tableau a été fixée à 50 éléments. Si le tableau est plein et que l'on demande l'ajout, une exception sera envoyée. On peut donc ajouter des éléments, les supprimer, trier le tableau grâce au tri à bulles et enfin afficher le panier avec sa méthode `ToString()`.