Rapport du Projet: “Le rhum de Guybrush"

Par: Amours Yann, Abbest Paul, Varachaud Rémi

Sommaire :

Introduction:

Premièrement, ce projet qui nous as été imposé comporte de nombreux problèmes à résoudre, sous formes d'un programme en langage C#. Ces problèmes sont variés mais les 2 principaux, le codage et le décodage d'une carte stockée dans une fichier. Puis d'autres problèmes plus secondaires consistant à réaliser un moyen de pouvoir afficher le nombre de parcelles qu'une carte contient ou bien même la taille moyenne d'une parcelle d'une carte.

Nous devions alors résoudre ces différents problémes:

- Décodage d'une carte stockée dans un fichier .chiffre.
- Codage d'une carte contenu dans un fichier .clair, puis stocké dans un fichier .chiffre.
- L'affichage d'une carte claire (carte décodée).
- L'affichage de la liste des parcelles
- L'affichage de la taille d'une parcelle (une parcelle spécifique)
- L'affichage des parcelles dont la taille est supérieure à une borne donnée
- L'affichage de la taille moyenne des parcelles d'une carte claire.

Modélisation Objet de la solution Retenue:

La modélisation objet retenu pour ce projet est la suivante :

Une classe DecodageCarte.cs :

- Possède un constructeur DecodageCarte() prenant en paramètre le chemin d'accès de la carte à décoder. Lors de sa création le constructeur va lire le fichier et stocker les informations de la carte sous forme de tableau à 2 dimensions.
- Une méthode DecodageDeLaCarte() qui va permettre de décoder une carte qui à été stockée dans le tableau a 2 dimensions lors de la création de l'objet, et donc créer la nouvelle carte décodée elle aussi dans un tableau à 2 dimensions, elle va aussi initialiser la méthode InitParcelle().
- Une méthode Affiche() qui va permettre d'afficher la carte décodée.
- Une méthode InitParcelle() qui va permettre d'initialiser en mémoire un dictionnaire de données qui est indispensable pour d'autres méthodes de la classe.
- Une méthode Parcelles(), qui permet d'afficher la liste des parcelles en fonction de leurs lettres, et les coordonnées de chaque unité, ainsi que le total d'unités par parcelle.
- Une méthode TailleParcelle(char parc), qui prend en paramètre la lettre de la parcelle via le "char parc" et va afficher la taille de la parcelle de ce caractère.
- Une méthode TailleSupp(int nb), qui prend en paramètre un chiffre qui va venir afficher toutes les parcelles qui ont un taille supérieur à ce chiffre.

- Une méthode AireMoyenne(), qui va venir afficher l'aire moyenne d'une parcelle.

Une classe CodageCarte.cs :

- Un constructeur CodageCarte(string accesFichier), qui prend en paramètre le lien du fichier afin de lire celui-ci et de le stocker dans un tableau à 2 dimensions.
- Une méthode CodageDeLaCarte(), qui va venir coder la carte décodée, et stocker ces données dans un tableau à 2 dimensions.
- Une méthode CreerFichier(), qui va créer un fichier pour une carte codée.

Conclusion :

Apports du projet :

- Nous pensons que ce projet nous a permis de mieux se rendre compte de ce qu'était un travail en équipe. En effet, nous avons bien réussi à nous répartir les tâches puisque lorsque quelqu'un codait, les deux autres l'aident sur Discord via un partage d'écran. De cette manière, tout le monde a participé, et il n'y a pas eu de déséquilibre.
- Nous avons aussi pu apprendre les uns des autres, que ce soit en tant que personne, mais aussi au niveau des connaissances du langage. En effet, certains avaient plus de connaissances et cela nous a permis d'apprendre de nouvelles choses tel que les Dictionary par exemple.
- Ce projet a aussi permis à l'équipe d'apprendre à utiliser GitHub qui est un outil vraiment puissant et efficace pour les travaux en équipe avec par exemple son intégration du "Merge".

Difficultés rencontrés :

- Lors de la programmation de la partie "Décodage" de la carte, nous avons rencontré des difficultés à mettre en place un algorithme prenant en compte toutes les possibilités sans faire un code qui soit redondant. Finalement, après une longue réflexion, nous pensons avoir réussi à éviter la redondance.

Éventuels axes d'amélioration du programme :

- Nous pensons que la partie Décodage de la carte peut probablement être un peu améliorée

Dépôt GIT:

https://github.com/Raviiiing/Projet-CS/tree/master
(Faire attention de bien être dans la branch “Master” et non “dev”)

Contributeurs:

- Amours Yann (Raviiiing).
- Abbest Paul (Shayzette).
- Varachaud Rémi (R3VaRa).