

Projet de NSI : Code Breaking

I- Contexte du projet

Dans le cadre de la spécialité NSI, notre groupe composé de Romain, Louis et Nael a dû élaborer un projet et l'exécuter. Ce projet était la création d'un jeu, ayant pour principe le jeu « Code Breaking ». Le but de ce projet était d'améliorer nos compétences en programmation en python ainsi que le travail en groupe.

II- Présentation du projet

Ce jeu se présente sous la forme d'un code python et d'une interface graphique. Les règles du jeu sont simples : un joueur doit deviner un code couleur choisi par l'ordinateur. Pour cela, il a le droit à un nombre d'essais limités. Dans ces essais, le joueur essaye de deviner le code à l'aide de boutons et l'ordinateur lui dit si les couleurs choisies sont bien placées, mal placées ou tout simplement pas dans le code. Si le joueur arrive à trouver le bon code couleur dans le nombre d'essais imparti, il gagne. Le joueur gagne des récompenses en gagnant. Il gagne des points proportionnellement à la difficulté de la partie et au nombre d'essais. Les points servent à débloquer des cosmétiques.

III- Problèmes à résoudre

Au début du projet, nous pensions que le code qui détermine si les couleurs sont bien ou mal placées allait être simple à programmer cependant nous avons rencontrés quelques difficultés tel que la fausse multiplicité des erreurs (2 mal placées au lieu de 1).

IV- Résolution des problèmes

Ce problème a été résolu de façon simple en faisant un brouillon afin de mieux comprendre le problème pour pouvoir le résoudre.

<p>Fonctionnement</p> <p>Le programme du jeu fonctionne avec 4 fonctions principales logiques.</p>
<p><code>NouvellePartie</code> (parametre) type : <code>partie</code> créer une partie en fonction de paramètres</p> <p>Cette fonction permet de créer une nouvelle partie pour le joueur, créée selon les paramètres configurés par le joueur. Elle créer une nouvelle grille vierge en fonction des dimensions spécifiés. Elle créer un code secret respectant les règles soumises.</p>
<p><code>correction</code> (ligne, code, parametres) type : <code>list</code> retourne la correction d'une partie</p> <p>Cette fonction a pour rôle d'indiquer au joueur les jetons biens placés et mal placés. Elle prend en compte les paramètres de la partie, pour savoir si le mode apprentissage est activé. Ce mode corrige le joueur en indiquant la position des jetons biens placés et mal placés.</p>
<p><code>possibilite</code> (parametres) type : <code>int</code> retourne le nombre de combinaisons possibles de code.</p> <p>Cette fonctions retourne le nombre de possibilités de code possibles selon les paramètres d'une partie. Le calcul se fait selon des notions vu en math en probabilités. On trouve par : $\text{nombre de jeton}^{\text{nombre de colonne}}$ avec remise des jetons.</p> <p>Ou avec : $\prod_{n=0}^{\text{colonnes}} (\text{jetons} - n)$</p> <p>si il n'y a pas de doublon dans le code secret.</p>
<p><code>difficulte</code> (parametres) type : <code>int</code> retourne la difficultés et les gains si victoire.</p> <p>Retourne la difficultés de la partie selon le nombre de combinaisons de code possibles :</p> <p>(dans le cas avec tout code possibles) $\frac{\text{nombre de possibilités}}{(\text{nombre d'essai})^2}$</p> <p>Relation pour les codes à couleurs uniques :</p> $\frac{\text{nombre de possibilités}}{(\text{nombre d'essai})!}$ <p>Le résultat est ensuite ramené à un sol et un plafond de 0 à 100. La difficultés de la partie correspond au gain du joueur si victoire.</p>

V- Membres du groupe

Le groupe se compose de 3 membres, Romain, Louis et Nael. Les deux premiers se sont concentrés sur la programmation du code python avec « une aide de Nael » tandis que ce dernier s'est concentré sur « l'élaboration » avec les conseils et les impressions de Romain et Louis.

VI- Voies d'amélioration

Dans ce projet, nous nous sommes grandement améliorés, car nous avons développé le projet à un stade pas encore atteint auparavant. Cela a permis de nous rendre compte que nous pouvons désormais créer un jeu avec à peu près toutes les fonctionnalités disponibles avec une interface graphique. Nous avons beaucoup poussé cette partie, notamment avec la customisation des règles et de certains graphiques.

VII- Progrès réalisés par chacun

Nael : Lors de ce projet, j'ai fortement amélioré mes compétences en rédaction lors de la création du compte rendu ainsi qu'en travail d'équipe lors de la répartition des tâches.

Romain : J'ai appris à utiliser les classes pour mieux optimiser le code. Il me reste à optimiser les graphismes pour ne pas laguer.

Louis : Dans ce projet, j'ai beaucoup appris, car Romain avait déjà fait des projets plus avancés que moi et il était en mesure de m'apprendre les fonctions graphiques de pygame. Maintenant je comprends bien ce milieu.

Conclusion

Le résultat final est réussi, tous nos objectifs ont été réalisés. Nous avons appris à travailler ensemble et en communication pour apprendre des connaissances de chacun. Nous avons découvert la collaboration de points de vue des ressources graphiques. Nous envisageons de continuer ensemble des défis dans le cadre d'internet, comme par exemple des games jamaes sur itch.io