

1 Contexte

En ce début 2022, le monde du jeu numérique a été marqué par plusieurs annonces majeures dont l'acquisition par Microsoft d'Activision Blizzard pour près de 70 milliards de dollars. Cependant, la nouvelle qui a fait le plus de buzz, est l'acquisition par le New York Times du célèbre jeu WORDLE. Cette acquisition a généré une question existentielle pour des millions de joueurs quotidiens : WORDLE va-t-il rester libre ?

C'est sur ce jeu et cet univers que porte le projet pluridisciplinaire du second semestre cette année.

2 WORDLE

L'application WORDLE est d'une simplicité d'usage étonnante (ce qui est certainement la raison première de son succès). L'objectif est de trouver un mot du dictionnaire (de 5 lettres par défaut et 6 essais, mais le jeu est paramétrable) caché par l'ordinateur. Pour cela, le joueur saisit à chaque étape un mot (qui existe dans le dictionnaire) de la même longueur que le mot à trouver. En réponse à cette proposition, l'ordinateur indique pour chaque lettre saisie :

- si elle est dans le mot recherché et si elle est à la bonne place,
- si elle est dans le mot recherché mais pas à place que vous avez choisie,
- si elle n'est pas dans le mot recherché.

Pour informer le joueur du résultat, un code de couleurs est utilisé : vert, jaune, gris (voir illustration ci-dessous).

Si vous avez trouvé le mot, le jeu s'arrête.

Si vous n'avez pas trouvé le mot, vous pouvez en saisir un nouveau jusqu'à ce que vous atteignez le nombre maximal d'essais autorisés (un paramètre du jeu, 6 par défaut pour des mots de 5 lettres).

Si vous dépassez le nombre maximal d'essais, vous avez perdu.

A	R	I	S	E
R	O	U	T	E
R	U	L	E	S
R	E	B	U	S

FIGURE 1 – Un exemple de grille du jeu WORDLE résolu en 4 étapes (source wikipedia.org)

3 Objectif

Votre objectif est de sauver le monde des joueurs WORDLE en élaborant votre propre instance du jeu, mais pas uniquement. Vous construirez également un solveur WORDLE, i.e. un programme qui se prend pour un utilisateur et qui résout les puzzles WORDLE.

L'application WORDLE de base résultante doit :

1. être réalisée sur une architecture Python/Web/Base de données (technologies vues au premier semestre),
2. être réalisable dans les délais du projet,
3. être paramétrable par le joueur (longueur des mots, nombre maximal d'essais),
4. sauvegarder dans une base de données les parties jouées

L'application web comptera pour 30% de la note.

Les plus téméraires pourront ajouter des fonctions de gestion des utilisateurs, de sauvegarde des jeux par joueur, des replays, des mécanismes de récompense, de l'apprentissage de stratégies des joueurs, ... De telles fonctionnalités ne sont pas obligatoires. Elles permettront d'obtenir des points bonus mais ne peuvent remplacer les points du solveur décrit ci-dessous.

Le solveur WORLDE doit :

1. être implémenté exclusivement en langage C,
2. utiliser les structures de données les plus avancées pour être le plus efficace (en mémoire et/ou en temps) dans la résolution,
3. disposer d'un mode interactif permettant de suivre à l'exécution en pas à pas, son cheminement.

Le volet portant sur les structures de données étudiées sera prédominant dans l'évaluation de la qualité du solveur.

Les plus téméraires implémenteront s'ils le souhaitent, des stratégies pour permettre à leur solveur d'être plus performant. Nous fournissons en fin de sujet, quelques liens sur des sites proposant des stratégies (plus ou moins efficaces).

L'interactivité à mettre en œuvre sur le solveur est la suivante :

- au lancement de votre programme, celui-ci devra lire dans un fichier wself.txt la longueur des mots à traiter,
- le solveur affiche à chaque étape sur l'entrée/sortie standard, le mot qu'il propose,
- le joueur lui indique en réponse toujours via l'entrée standard (interaction dans le terminal) :
 - soit sa volonté d'arrêter le jeu (en saisissant la valeur -1),
 - soit le résultat de la proposition sous forme d'une séquence contenant les chiffres 0, 1 ou 2. Le chiffre 0 signifie que la lettre proposée ne se trouve pas dans le mot recherché ; 1 signifie que la lettre se trouve dans le mot recherché mais pas à cette place et 2 signifie que la lettre proposée est bien dans le mot cible et à sa place.

Par exemple, si le mot recherché est TALUS et que le solveur propose RABOT, la séquence à donner au solveur par l'utilisateur sera 02001 (R n'est pas dans le mot recherché, A est bien à sa place, B n'est pas dans le mot recherché et T est dans la cible mais pas à sa place).

Le solveur représentera 50% de la note.

Pour les deux applications, vous chercherez à supporter l'ensemble des mots du dictionnaire français (ou d'une autre langue si vous souhaitez le rendre multi-langues). Une recherche web dans ce sens vous permettra de trouver facilement des données pour cela.

La composante Gestion de Projet intégrant rapport et soutenance compte pour 20% de la note. Celle-ci doit donc être mise en œuvre de façon systématique, méticuleuse et traçable. Une note inférieure à 10/20 dans l'une des 3 composantes invalide le projet.

4 Conception de votre application

À partir de l'état de l'art, vous allez concevoir une application qui réalise un service.

La conception détaillée d'une application informatique est un processus long et méticuleux, s'appuyant sur des principes, méthodes, concepts et outils que vous ne maîtrisez pas encore (vous apprécierez d'autant plus les cours qui y seront consacrés dans les semestres à venir).

Ce qui est attendu sous le terme conception pour ce projet est :

- une présentation de WORDLE,
- une présentation textuelle des deux applications, la présentation argumentée du schéma de la base de données, l'enchaînement des pages ainsi que les "mockups", les paramétrages du jeu,
- une description détaillée, motivée et argumentée des structures de données utilisées dans le solveur,

Ce document de conception doit impérativement faire l'objet d'une validation par l'équipe de gestion du module avant tout démarrage de l'implémentation !. Cette validation se fait au fil de l'eau à la demande des groupes.

4.1 Implémentation

En vous répartissant les tâches préalablement définies dans le groupe, tout en vous assurant que chacune et chacun fournit de contributions sur les 2 applications cibles : le jeu et le solveur.

4.2 Complexité, tests et validation

▷ **Question 1:** Pour les fonctions majeures réalisées dont la complexité n'est ni linéaire, ni constante, vous donnerez l'analyse en complexité et en mémoire, tout particulièrement pour le solveur.

▷ **Question 2:** Vous détaillerez les implémentations et tests des structures de données avancées utilisées pour le projet, tout particulièrement sur le solveur.

5 Organisation

Le projet se fait par groupes de 4 élèves librement constitués. Chaque groupe envoie au responsable du module (Olivier.Festor@telecomnancy.eu) la liste de ses membres et en mettant en copie tous les membres du groupe.

Dès réception des groupes, l'école affectera à chaque groupe un dépôt gitlab dédié sur lequel devront être déposées toutes les productions du groupe (code, tests, documents source, graphes de performance, illustrations, éléments de gestion de projet, ...). Le dépôt et son contenu serviront de référence pour la notation du projet. L'étude de l'usage régulier de celui-ci par l'ensemble des contributeurs ainsi que l'analyse fine des contenus validés sont également des éléments importants de suivi de la dynamique du projet et d'identification des contributions des membres d'un groupe.

6 Déroulé et livrables du projet

L'objectif premier de ce projet est de vous amener à consolider et mettre en œuvre vos compétences en langage C, en structures de données ainsi qu'en gestion de projets. Les connaissances et compétences seront acquises au travers de la recherche de documentation technique et scientifique, l'étude d'algorithmes, de structures de données, leur codage, leur test et leur livraison au client, le tout en travaillant en équipe et en appliquant les principes appris en cours.

Chaque membre du groupe doit dans le projet, développer des compétences dans tous les champs du projet : (1) gestion de projet, (2) état de l'art et compréhension d'algorithme, (3) développement de fonctions avancées en C, (5) documentation et (6) présentation orale des résultats et des démonstrations. Tout membre d'un groupe doit notamment pouvoir expliquer toutes les structures de données et les algorithmes du projet et tous les codes associés.

Les livrables du projet sont :

- l'ensemble des fichiers sources des implémentations,
- un rapport rédigé en LaTeX synthétisant le travail et comprenant en plus d'une introduction et d'une conclusion :
 - trois sections consacrées à la conception et à l'implémentation des différentes parties de vos applications ;
 - une section consacrée aux tests et aux performances ;
 - une section dédiée à la gestion de projets.
- une présentation suivant le plan du rapport et synthétisant le travail ;
- une démonstration de vos réalisations.

Vous fournirez également en annexe de votre rapport, l'ensemble des documents produits pour la bonne gestion de votre projet.

La soutenance, la démonstration et la réponse aux questions se fera pour chaque groupe sur un créneau de **30 minutes** réparties comme suit :

1. Présentation structurée (avec planches powerpoint ou beamer) : 6 minutes maximum
2. Démonstration des fonctions : 10 minutes maximum

3. Réponse aux questions : 10 minutes maximum
4. Délibération du jury : 4 minutes

Le respect des temps interviendra dans la notation. Nous vous conseillons donc vivement de bien préparer votre soutenance et de vous assurer que vos programmes tournent sur les machines qui vous serviront à la présentation.

Vous n'êtes pas obligés d'atteindre tous les objectifs du projet ! Privilégiez la progression homogène sur tous les volets : compréhension des algorithmes, implémentation, documentation, test, mesures algorithme par algorithme plutôt que de vous précipiter sur le développement de toutes les fonctions.

N'hésitez pas à solliciter l'équipe pédagogique si vous avez des questions !

7 Dates importantes

- Date de dépôt des groupes : **18 mars 2022, 18 :00 CEST**
- Date de rendu de projet (fermeture des repositories sur le serveur gitlab de l'école) : **31 mai 2022, 23 :00 CEST**
- Date de soutenance : **10 juin 2022**

8 Travail personnel et honnêteté scientifique

Ne trichez pas ! Ne copiez pas ! Si vous le faites, vous serez lourdement sanctionnés. Nous ne ferons pas de distinction entre copieur et copié. Vous n'avez pas de (bonne) raison de copier. En cas de problème, nous sommes prêt à vous aider. Encore une fois : en cas de doute, contactez vos enseignants. Vous ne les dérangez pas !

Par tricher, nous entendons notamment :

- Rendre le travail d'un collègue en y apposant votre nom ;
- Obtenir un code, une solution par un moteur de recherche et la rendre sous votre nom ;
- Récupérer du code et ne changer que les noms de variables et fonctions ou leur ordre avant de les présenter sous votre nom (*"moving chunks of code around is like moving food around on your plate to disguise the fact that you haven't eaten all your brussel sprouts"*) ;
- Autoriser consciemment ou inconsciemment un collègue à s'approprier votre travail personnel. Assurez vous notamment que votre projet et ses différentes copies locales ne soient lisibles que par vous et les membres de votre groupe.

Nous encourageons des séances de brainstorming et de discussion entre les élèves sur le projet. C'est une démarche naturelle et saine comme vous la rencontrerez dans votre vie professionnelle. Si les réflexions communes sont fortement recommandées, vous ne pouvez rendre que du code et des documents écrits par vous-même. Vous indiquerez notamment dans votre rapport toutes vos sources (comme les sites internet que vous auriez consultés), en indiquant brièvement ce que vous en avez retenu.

Il est quasi certain que nous détectons les tricheries. En effet, les rapports et les codes sont systématiquement soumis à des outils de détection de plagiat et de copie. Il existe notamment un outil de détection de manipulation de code extraordinaire mis à disposition par l'Université de Stanford (MOSS). De plus, chacun a son propre style de programmation, et personne ne développe la même chose de la même manière. .

9 Quelques liens utiles sur le jeu WORDLE

- un tutoriel pour jouer à WORDLE <https://www.youtube.com/watch?v=PXUqGT5ySsc>
- un tutoriel sur l'utilisation de la théorie de l'information pour optimiser la résolution d'une session Wordle <https://www.youtube.com/watch?v=v68zYyaEmEA&t=250s>
- le complément du tutoriel précédent qui y corrige un bug <https://www.youtube.com/watch?v=fRed0Xmc2Wg>