

Projet : Boggle

1 Boggle :

Le jeu prend en général la forme d'une grille de taille 4×4 composée de 16 lettres. Un compte à rebours de 3 minutes est lancé et tous les joueurs commencent à jouer.

Chaque joueur cherche des mots pouvant être formés à partir de lettres adjacentes du plateau. Par « adjacentes », il faut comprendre horizontalement (de gauche à droite ou de gauche à droite), verticalement (de haut en bas ou de bas en haut), ou en diagonale (en montant ou en descendant). Les mots doivent être de 3 lettres au minimum, peuvent être au singulier ou au pluriel, verbes conjugués ou non, mais on ne peut utiliser plusieurs fois la même case pour le même mot. Les joueurs écrivent tous les mots qu'ils ont trouvés sur leur feuille personnelle. Après les 3 minutes de recherche, les joueurs doivent arrêter d'écrire et on calcule le total des points.

Celui qui a le plus de points gagne et plus les mots sont longs, plus ils rapportent de points :

- 3 et 4 lettres : 1 point
- 5 lettres : 2 points
- 6 lettres : 3 points
- 7 lettres : 5 points
- 8 lettres et plus : 11 points

Pour être intéressantes, les grilles de jeu doivent refléter la répartition des diverses lettres dans la langue française (les lettres 'Q' ou 'K' apparaissent peu...). On pourra par exemple utiliser le système de pondération du Scrabble.

Il existe une version en ligne à l'adresse <http://massiveboggle.fr/> idéale pour vous familiariser avec le jeu.

2 Méthode de résolution :

Pour être accepté, un mot doit être présent dans le dictionnaire et bien sûr être de longueur au moins égale à 3. Un dictionnaire est fourni sous forme de fichier texte : `complet.txt`

3 Travail à réaliser :

Catégories de programmes : le programme que vous allez rendre sera classé dans l'une des catégories ci-dessous et sera noté par rapport à la note maximale associée.

– **Version 1 : Solveur de boggle**

La grille a la forme d'un rectangle dont les dimensions sont modifiables par l'utilisateur.

On génère une grille aléatoire (tenant compte des fréquences d'apparition des lettres dans la langue française) et on détermine l'ensemble des mots présents à la fois dans le dictionnaire (fichier texte fourni) et dans la grille avec les coordonnées de leur première lettre.

Ces mots sont affichés par longueur croissante et par ordre alphabétique à côté de la grille. L'affichage est en mode texte.

– **Version 2 : jeu de boggle**

L'utilisateur peut jouer à résoudre une grille de boggle : la saisie des mots se fait au clavier ; au fur et à mesure de leurs saisies, la machine répond s'ils sont acceptés ou non. A la fin du jeu, la machine affiche par longueur croissante et par ordre alphabétique la liste des mots oubliés par le joueur et le score maximal possible.

– **Version 3 : Jeu de boggle : version conviviale**

Les affichages et saisies utilisent la librairie graphique ncurses.

La documentation de cette librairie est disponible en ligne principalement en langue anglaise. Implémenter une aide possible : par exemple, lorsque vous sélectionnez une lettre, la machine vous indique la longueur du mot le plus long (non encore trouvé) qui commence par les lettres sélectionnées.

Attention : Pour obtenir les points relatifs à une version, il faut que les versions inférieures aient été traitées dans leur intégralité.

4 Notation

Le programme que vous allez rendre sera classé dans l'une des catégories détaillées ci-dessus et sera noté par rapport à la note maximale associée. Dans les 3 versions vous devez utiliser divers modules (-h et .c) que vous compilerez séparément avec un fichier makefile adapté (utilitaire make). Selon la version choisie, vous êtes notés sur :

- version 1 : 6 points
- version 2 : 8 points
- version 3 : 10 points

Les critères de notation sont : efficacité du code (efficacité des algorithmes), forme du code (lisibilité, commentaires bien choisis, présence d'en-têtes de fonctions et de modules, conventions de présentation homogènes...), présence de fichiers de tests unitaires et de fichiers de tests fonctionnels. Cette notation sera faite principalement lors de la recette qui aura lieu pendant la dernière séance de TP.

Les 10 points restant sont relatifs à la partie gestion de projet.

5 Règles de codage minimales

- Au niveau des modules :

Chaque module de l'application débutera par un entête. Vous définirez un modèle d'entête dans lequel devront figurer les informations que vous jugerez nécessaires pour une bonne

compréhension du rôle de ce module.

Exemple : le ou les noms des auteurs, la date de création du fichier, la date et le pourquoi de la dernière modification ...

Chaque nom de fonction appartenant à un même module sera préfixé par une même chaîne de caractères symbolisant le module.

- Au niveau des fonctions :

Chaque fonction sera précédée par un commentaire de description. Vous définirez un modèle de commentaire dans lequel devront figurer les informations que vous jugerez nécessaires pour une bonne compréhension du rôle de la fonction.

Exemple : nom de la fonction, liste des paramètres, rôle et type(entrée, sortie) de chaque paramètre.

Les commentaires figurant dans le corps de la fonction devront être pertinents et placés en fonction des différentes étapes de l'algorithme associé.

Choisissez des règles de nommage des variables en fonction du statut de ces variables (locales à une fonction, à un module, externes).

6 Les fichiers de tests

1. Tests unitaires : **Un fichier de test pour chaque module** *test_u_nom_module.c* contenant un main appelant les diverses fonctions du module à tester.
2. Tests fonctionnels : Un fichier *test_fonctionnel.c* qui permet de tester le logiciel avec un jeu de grilles prédéfinies.

On pourra placer chacun de ces tests dans un sous-répertoire du répertoire courant et créer un makefile associé à chaque test.

