Le jeu des Mots Mêlés

Préambule

Ce travail est à réaliser seul ou en binôme du même groupe

Il est à rendre au plus tard à la fin de votre dernière session, semaine qui précède les vacances de Noël

Vous déposerez votre solution Visual Studio sur DVO au format .zip sous votre ou vos noms

Attention à ne pas déposer une partie de votre solution, vérifiez que vous avez intégré tous les fichiers nécessaires.

Un programme ne compilant pas ou ne s'exécutant pas entraîne une note de 0/20.

L'ensemble des codes sera analysé par un système anti-plagiat. Un plagiat entraîne une note de 0/20 au module (pour les 2 protagonistes).

Le sujet proposé a pour objectif de mettre en application tous les concepts vus en TD, c'est pourquoi il est impératif de suivre les énoncés tels qu'ils ont été écrits.

La dernière séance donnera lieu à une revue de code. Un étudiant doit être en mesure de décrire n'importe quelle partie de code (y compris une portion écrite par son binôme).

Recommandations générales

Ne pas oublier votre projet Test Unitaire sur au moins le test de 5 fonctions.
Ne pas oublier les commentaires ///
Ne pas oublier d'écrire les variables et fonctions avec des noms lisibles
Ne pas oublier l'indentation.

Un ensemble de méthodes sont imposées pour faciliter une correction précise

Vous pouvez rajouter d'autres méthodes si les besoins de votre code l'imposent évidemment.

Ne pas oublier que l'utilisation de

```
Random  r = new Random()
```

ne peut se faire qu'une seule fois. Ensuite r.Next(..) peut se faire autant de fois que nécessaire

Présentation du problème

Le jeu met en compétition plusieurs joueurs sur une grille de mots.

Le joueur a une grille de mots cachés qu'il doit trouver dans le temps imparti (1mn) pour le premier tour. Il remporte un bonus dans le cas positif et sinon remporte un score égal aux nombres de lettres des mots trouvés.

Le joueur suivant fera de même sur une **nouvelle** grille. Il est difficile de jouer la simultanéité sur ce problème.

A chaque tour, la dimension de la grille et le nombre de mots cachés augmentent ainsi que la difficulté et le temps imparti.

Celui qui a gagné est celui qui sera le plus rapide pour trouver tous les mots cachés ou qui aura le score le plus élevé.

Exemples

Cas le plus simple : (donné en pièce jointe)

La grille est composée de 7 lignes et 6 colonnes.
Les 8 mots à trouver sont sur les lignes (de gauche à droite) et les colonnes (de haut en bas)

1	7	6	8				
AGILITE	BUS	ETOILE	ROUTE	SENS	VIE	VOITURE	VOYAGE
M	V	E	D	A	S		
V	O	Y	A	G	E		
I	I	T	N	I	S		
E	T	O	I	L	E		
A	U	V	B	I	T		
I	R	O	U	T	E		
S	E	N	S	E	P		

Les mots à trouver sont : Voyage, Vie, Etoile, Route, Sens, Agilité, Bus, Voiture,

AGILITE	BUS	ETOILE	ROUTE	SENS	VIE	VOITURE	VOYAGE
M	V	E	D	A	S		
V	O	Y	A	G	E		
I	I	T	N	I	S		
E	T	O	I	L	E		
A	U	V	B	I	T		
I	R	O	U	T	E		
S	E	N	S	E	P		

Cas le plus complexe : (donné en pièce jointe)

La grille est composée de 13 lignes et de 13 colonnes, 28 mots sont à trouver

Les mots sont sur les lignes, les colonnes et les diagonales écrits dans tous les sens …

13	13	28														
AGRICULTUR	ALIMENTATI	BREBIS	CANARD	CHEVRES	CIDRE	CONFITURE	ELEVEUR	ESCARGOT	ESTIVE	FERME	FROMAGE	GASCONNE	GOUT	MARAICHER	MAR
T	E	R	R	E	R	U	T	I	F	N	O	C			
I	S	O	C	T	L	N	Z	R	E	P	Q	C			
U	C	C	I	N	D	E	O	E	R	O	U	E			
D	A	H	D	E	R	M	V	A	M	M	A	R			
O	R	E	R	V	A	S	A	E	E	M	L	U			
R	G	V	E	G	N	N	V	R	U	E	I	T			
P	O	R	E	H	A	E	L	E	C	R	T	L			
S	T	E	S	E	C	S	M	A	R	H	E	U			
I	N	S	T	T	L	U	T	U	O	G	E	C			
B	I	F	I	A	G	T	R	U	I	T	E	I			
E	A	U	V	M	A	R	A	I	C	H	E	R			
R	S	E	E	O	E	N	N	O	C	S	A	G			
B	N	O	I	T	A	T	N	E	M	I	L	A			

Au début de la partie, vous proposez au joueur une grille d'une difficulté précise

- à partir d'un fichier déjà créé
- ou bien une grille que vous générez aléatoirement (que vous sauvegarderez dans un fichier .csv ainsi que les mots générés aléatoirement selon la structure préconisée ci-après)

Vous démarrez alors un timer pour le joueur.

Lorsque le joueur joue, il doit saisir un mot, sa direction (N,S,E,O,NE,NO,SE,SO) et le point d'ancrage du mot (ligne et colonne)

Le mot proposé doit évidemment exister dans le dictionnaire ci-joint. Chaque mot a au moins 2 lettres. Vous pouvez jouer dans une langue ou une autre. Deux dictionnaires vous sont proposés en Français et en Anglais

```
2
AA AH AI AN AS AU AY BI BU CA CE CI DA DE DO DU EH EN ES ET EU EX FA FI GO HA HE HI HO IF IL IN JE KA LA LE LI LU MA ME MI MU NA NE NI NO NU OC OH ON
3
AAS ACE ADA ADO AGA AGE AGI AIE AIL AIR AIS AIT ALE ALU AME AMI ANA AND ANE ANS API ARA ARC ARE ARS ART ASA ASE AUX AVE AXA AXE AYS BAC BAH BAI BAL BA
KWA KYU LAC LAD LAI LAO LAS LEI LEK LEM LES LET LEU LEV LEZ LIA LIE LIN LIS LIT LOB LOF LOG LOI LOS LOT LUE LUI LUS LUT LUX LYS MAC MAI MAL MAN MAO MA
TRI TUA TUB TUE TUF TUS TUT UNE UNI UNS URE USA USE UTE VAL VAN VAR VAS VAU VER VES VET VIA VIE VIF VIL VIN VIS VIT VOL VOS VUE VUS WAX WEB WOK WON WL
4
ABAT ABBE ABRI ABUS ACES ACRE ACTE AERA AERE AFIN AGEE AGES AGIO AGIR AGIS AGIT AIDA AIDE AIES AIGU AILE AIMA AIME AINE AIRE AIRS AISE AISY AJUT ALEA
CHAS CHAT CHEF CHER CHEZ CHIC CHOC CHOU CHUT CIEL CILS CIME CINQ CIRA CIRE CITA CITE CLAM CLAN CLAY CLEF CLES CLIN CLIP CLOS CLOT CLOU CLUB COCO CODA
FARD FARS FART FATS FAUT FAUX FAXA FAXE FEES FELA FELE FEND FERA FERS FERU FETA FETE FETU FEUE FEUX FEVE FIEE FIEF FIEL FIER FIES FIEZ FIGE FILA FILE
JABS JACK JACO JADE JAIN JAIS JALE JAMS JANS JARD JARS JASA JASE JASS JAVA JAZZ JEAN JEEP JERK JETA JETE JETS JEUN JEUX JEZE JOBS JOIE JOJO JOLI JONC
MEME MENA MENE MENS MENT MENU MERE MERS MESS METS MEUS MEUT MEZE MICA MIDI MIEL MIEN MILE MIMA MIME MINA MINE MIRA MIRE MISA MISE MISS MITA MITE MIXA
PIED PIES PIEU PIGE PILA PILE PINS PION PIPE PIRE PISE PLAN PLAT PLIA PLIE PLIS PLOT PLUS PLUT PNEU POIL POIS POIX POLE POLI POLO POLY POND PONT POOL
SIEN SIKH SILO SILY SIRE SISE SITE SKAI SKAS SKAT SKIA SKIE SKIF SKIN SKIP SKIS SKUA SLIP SLOW SNOB SOCS SODA SOFA SOIE SOIF SOIN SOIR SOIS SOIT SOJA
VIOL VIRA VIRE VISA VISE VITE VIVE VOEU VOIE VOIR VOIS VOIT VOIX VOLA VOLE VOLS VOLT VOMI VONT VOTA VOTE VOUA VOUE VOUS VRAC VRAI VUES WALE WALI WASP
5
ABATS ABBES ABCES ABIMA ABIME ABOIE ABOIS ABOLI ABORD ABOYA ABOYE ABRIS ABUSA ABUSE ABYME ACCES ACCRU ACERE ACHAT ACIDE ACIER ACRES ACTES ACTEZ ACTIF
E ARMER ARMES ARMEZ AROME ARQUA ARQUE ARRET ARYEN ARYLE ARZEL ASILE ASPIC ASQUE ASSEZ ASSIS ASSIT ASTRE ATHEE ATLAS ATOLL ATOME ATONE ATOUT ATRES AUCL
IGE BEKES BELER BELEZ BELGE BELLE BELON BEMOL BENET BENIE BENIN BENIR BENIS BENIT BENJI BENNE BERCA BERCE BERET BERGE BERNA BERNE BERYL BETES BETON BE
BUCHE BUEES BUGGY BULBE BULLE BUQUA BUQUE BURIN BUSES BUSEZ BUSTE BUTEE BUTER BUTES BUTEZ BUTIN BUTOR BUTTE BUVEZ BYTES CABAN CABAS CABLA CABLE CABRA
E CHYME CIBLE CIDRE CIEUX CIGUE CIMES CIREE CIRER CIRES CIREZ CITEE CITER CITES CITEZ CIVET CIVIL CLAIE CLAIR CLAMA CLAME CLAMS CLANS CLARK CLAYS CLEF
RER CURES CUREZ CURRY CUVEE CUVER CUVES CUVEZ CYANS CYCAS CYCLE CYCLO CYGNE CYMES CYONS CZARS DADAS DAGUE DAIMS DAKIN DALLA DALLE DAMER DAMES DAMEZ DA
DOUZE DOYEN DRAIN DRAME DRAPA DRAPE DRAPS DRAYA DRAYE DRINK DROIT DROLE DROPS DRUES DRUZE DUAUX DUCAL DUCHE DUELS DUNES DUNKS DUPEE DUPER DUPES DUPEZ
R ESTOC ETAGE ETAIE ETAIN ETAIS ETAIT ETALA ETALE ETALS ETANG ETANT ETAPE ETATS ETAUX ETAYA ETAYE ETEND ETHER ETIEZ ETIRA ETIRE ETOLE ETRES ETRON ETUI
NIE FINIR FINIS FINIT FIOLE FIONS FIRME FISCS FIXAI FIXAS FIXAT FIXEE FIXER FIXES FIXEZ FJELD FJORD FLAIR FLANA FLANC FLANE FLANS FLAPI FLASH FLEAU FL
GARER GARES GAREZ GARNI GATEE GATER GATES GATEZ GAULE GAVEE GAVER GAVES GAVEZ GAYAL GAZAI GAZAS GAZAT GAZEE GAZER GAZES GAZEZ GAZON GAZOU GEAIS GEANT
A HALEE HALER HALES HALEZ HALLE HALLS HALOS HALTE HAMAC HAMPE HANTA HANTE HAPAX HAPPA HAPPE HARAS HARDI HAREM HARKA HARKI HAROS HARPE HASES HATEE HATE
ONS JABLA JABLE JABOT JACEE JACKS JACOS JACOT JACTA JACTE JADES JADIS JAINA JAINE JAINS JALAP JALES JALON JAMBE JANTE JAPON JAPPA JAPPE JAQUE JARDE JA
KAREN KARMA KARST KARTS KATAS KAVAS KAWAS KAWIS KAYAC KAYAK KEBAB KEFIR KENDO KEPIS KERMA KETCH KEUFS KEUMS KHANS KHATS KHMER KHOIN KHOLS KICKS KIEFS
S LIEES LIEGE LIENS LIENT LIERA LIEUE LIEUX LIFTS LIGNE LIGUA LIGUE LIIEZ LILAS LIMEE LIMER LIMES LIMEZ LIMON LINGE LINKS LINOS LIONS LIPPE LIPPU LIRA
RNE MARRA MARRE MARRI MARTE MASSA MASSE MATCH MATER MATES MATEZ MATHS MATIN MATIR MAUVE MAXIS MAYAS MAYEN MAYES MAZAI MAZAS MAZAT MAZEE MAZER MAZES MA
MUERA MUETS MUFLE MUGIR MUGIS MUGIT MUIEZ MULES MULET MULOT MUNIE MUNIR MUNIS MUNIT MUONS MURAL MUREE MURER MURES MURET MUREX MUREZ MURIE MURIR MURIS
N OKAPI OLIVE OMBRE OMETS OMISE ONCES ONCLE ONDEE ONDES ONGLE ONLAY OPALE OPERA OPERE OPINA OPINE OPIUM OPTER OPTES OPTEZ ORAGE ORALE ORAUX ORDRE OREE
VES PAVEZ PAVOT PAXON PAYAI PAYAS PAYAT PAYEE PAYER PAYES PAYEZ PAYSE PEAGE PEAUX PECHA PECHE PEDZA PEDZE PEGRE PEINA PEINE PEINS PEINT PEKAN PEKET PE
```

A l'issue du temps imparti, vous établissez le score du joueur et le sauvegardez.

Le joueur suivant fait de même sur une autre grille de même difficulté.
Aux tours suivants, vous augmentez la difficulté : nombres de mots, puis une diagonale, puis 2 diagonales, puis lecture inversée des mots …

Vos algorithmes seront basés sur la programmation objet et pour cela vous créerez au moins 4 classes : Joueur, Plateau, Dictionnaire et la classe Jeu.

Le jeu peut être sauvegardé pour être repris à un autre moment. Même si le codage de la sauvegarde ou la reprise est un bonus, la réflexion sur les structures de données utiles à la sauvegarde du jeu sont imposées. Pensez donc aux structures de données utiles lors d'une sauvegarde du jeu complet pour une reprise ultérieure.

L'ordre des classes proposé n'est pas forcément celui de la mise en place de votre solution globale

Classe Joueur (à réaliser dans un fichier Joueur.cs)

Un joueur est caractérisé par son nom, par les mots trouvés et les scores par plateau au cours de la partie

La création d'un joueur n'est possible que si celui-ci a un nom au départ du jeu. Le score et les mots trouvés sont respectivement égal à 0 et null au départ du jeu.

Vous créerez les propriétés en fonction des besoins de votre programme

Les méthodes suivantes sont imposées : (les signatures peuvent être ajustées en fonction de votre code)

`public void Add_Mot (string mot) ajoute le mot dans la liste des mots déjà trouvés par le joueur au cours de la partie`

`public override string ToString() ou public string toString() qui retourne une chaîne de caractères qui décrit un joueur.`

`public void Add_Score(int val) qui ajoute une valeur au score`

Classe Dictionnaire (à réaliser dans un fichier Dictionnaire.cs)

Une instance de dictionnaire associe un ensemble de mots avec une longueur déterminée ainsi qu'une langue.

Les méthodes sont imposées : (les signatures peuvent être ajustées en fonction de votre code)

`public override string ToString() ou public string toString() qui retourne une chaîne de caractères qui décrit le dictionnaire à savoir ici le nombre de mots par longueur et la langue`

`public bool RechDichoRecursif(string mot) qui teste que le mot appartient bien au dictionnaire. Vous utiliserez cette méthode pour rechercher un mot dans le dictionnaire !`

Classe Plateau (à réaliser dans un fichier Plateau.cs)

Une instance de plateau est définie par une matrice, un niveau de difficulté et les mots à rechercher.

Deux cas doivent être pris en considération :

- Générée automatiquement et aléatoirement avec un degré de difficulté
- Qui s'initialise avec une instance de plateau définie à partir d'un fichier déjà existant.

Les fichiers .csv sont décrits de la manière suivante :

Première ligne Entête : niveau de difficulté, nombre de lignes, nombre de colonnes, nombre de mots à trouver

Deuxième ligne : les mots à trouver séparés par des ' ;'

Les lignes suivantes représentent le plateau chaque item de chaque ligne est séparé par un ' ;'

Nous définissons les niveaux de difficultés suivants :

- 1 : les mots sont situés sur les lignes de gauche à droite et sur les colonnes de haut en bas
- 2 : les mots répondent à la difficulté 1 et peuvent être lus de gauche à droite et de bas en haut
- 3 : les mots répondent à la difficulté 2 et peuvent être sur les diagonales (NE-SO) de haut en bas
- 4 : les mots répondent à la difficulté 3 et peuvent être sur les diagonales (NO-SE) de haut en bas
- 5 : les mots répondent à la difficulté 4 et peuvent être sur toutes les diagonales de bas en haut

Vous créerez les propriétés en fonction des besoins de votre programme

Les méthodes suivantes sont imposées : (les signatures peuvent être ajustées en fonction de votre code)

`public override string ToString()ou public string toString() qui retourne une chaîne de caractères qui décrit le plateau.`

`public void ToFile(string nomfile) qui sauvegarde l'instance du plateau dans un fichier en respectant la structure précisée ci-dessus`

`public void ToRead(string nomfile) qui instancie un plateau à partir d'un fichier`

`public bool Test_Plateau(string mot,int ligne, int colonne, string direction) qui teste si le mot passé en paramètre est un mot éligible aux positions ligne et colonne et dans la direction indiquée`

`Un mot est éligible si`

1. Il est bien placé sur le plateau en vertical ou horizontal ou en diagonale en fonction de la difficulté
2. Il appartient au dictionnaire

Classe Jeu (à réaliser dans un fichier Jeu .cs)

La classe Jeu possède entre autres les attributs suivants :

- Une structure de Dictionnaire
- Un plateau courant et les plateaux précédents
- Les joueurs

Le programme principal Main va donc à tour de rôle donner la main à un joueur puis à un autre pendant un laps de temps pour la partie à définir (6mn par exemple).
Chaque tour progresse en difficulté, sur des matrices de dimensions variables (à vérifier bien sûr)

Chaque joueur a X minutes maximum(voir la classe DateTime et TimeSpan) pour trouver tous les mots sur un plateau, X étant à définir au début du jeu.

A la fin du temps imparti ou du temps réalisé par le joueur, le joueur suivant prend la main.

Le jeu est terminé quand le temps de la partie est terminé ou que les plateaux successifs se sont déroulés (5 au maximum).

Vous pouvez arrêter le jeu au bout d'un laps de temps et sauvegarder l'instance du jeu dans des fichiers selon les critères proposés en mode lecture. Vous affichez les scores temporaires de chacun des joueurs.

BONUS

Vous proposerez la sauvegarde du jeu possible ainsi que sa restitution

Documents à rendre au dernier TD

Vous déposez sur DVO lors de votre dernière séance

- Votre solution zippée
- Le diagramme de classe qui tient compte de la problématique de la sauvegarde du jeu
- Et un document explicatif sur une page maximum

Vous pourrez lors de cette séance, compléter les commentaires ou votre projet Tests Unitaires ou votre document explicatif.