

Nom1 :

Nom2 :

## **Projet « Le Pendu »**

**/20**

### **Phase 1 : Jeu de base (→ 12h15)**

#### **Evaluation phase 1**

**/14**

**(A faire remplir par un professeur avant de passer à la phase 2)**

##### **Le code est fonctionnel**

Le jeu de base se déroule correctement

Les images et les commentaires sont appropriés

##### **Le code est bien écrit**

Les classes *FenetreJeuPendu* et *MotACompleter* sont bien utilisées

Les répétitives et les alternatives sont bien construites

##### **Le code est clair**

### **Phase 2 : Améliorations (→ 12h30)**

#### **Evaluation phase 2**

**/6**

Amélioration 1 : /2

Amélioration 2 : /2

Amélioration 3 : /2

**Le code est fonctionnel et reste clair malgré l'ajout de l'amélioration**

#### **Amélioration 1 :**

Si le caractère introduit n'est pas une lettre (minuscule ou majuscule), un message s'affiche et le joueur n'est pas pénalisé. (Le pendu n'augmente pas). Un message adéquat l'avertit.

La classe *Character* possède des méthodes intéressantes : `toUpperCase()`, `isLetter()`, ...

#### **Amélioration 2 :**

Si la lettre introduite a déjà été donnée, le joueur n'est pas pénalisé même si la lettre n'apparaît pas dans le mot. (Le pendu n'augmente pas). Un message adéquat l'avertit.

Vous allez utiliser une table de 26 booléens, un booléen pour chaque lettre de l'alphabet.  
Si le booléen est *true*, la lettre a déjà été lue, *false* sinon.  
(code unicode de 'A' → 65, code unicode de 'B' → 66, ...)

Exemple :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
F	F	F	F	T	F	F	F	T	F	F	F	F	T	F	F	F	T	F	F	F	F	F	F	F	F

Les lettres déjà lues → E I N R

Dans la classe *MotACompléter*:

Ajoutez l'attribut `lettresLues` (la table des 26 booléens).

Complétez Le constructeur.

N'oubliez pas de créer la table `lettresLues` (*new*) !

Rappel : lors de la création d'une table de booléens, tous les booléens sont à *false*.

Ajoutez la méthode `boolean dejaLue(char lettre)` qui renvoie le booléen associé à la lettre passée en paramètre dans la table `lettresLues`.

Cette méthode lance une exception de type *IllegalArgumentException* si le caractère passé en paramètre n'est pas une lettre.

Ajoutez la méthode `void ajouterLettreLue(char lettre)`. Cette méthode a comme effet de passer le booléen associé à la lettre passée en paramètre à *true*.

Cette méthode lance une exception de type *IllegalArgumentException* si le caractère passé en paramètre n'est pas une lettre.

Ensuite, modifiez la classe *JeuDuPendu*.

### **Amélioration 3 :**

Tout au long du jeu, les lettres déjà lues seront affichées. Les lettres utilisées apparaissent une seule fois et triées selon l'alphabet.

La classe *FenetrePendu* possède déjà une méthode `afficherLettresLues(String texte)` ! Utilisez-là pour les faire apparaître sur la fenêtre.

Commencez par ajouter, dans la classe *MotACompléter*, la méthode `String lettresLues()`. Cette méthode parcourt la table `lettresLues` pour créer une chaîne de caractères avec les lettres qui ont le booléen associé à *true*.

Ensuite, modifiez la classe *JeuDuPendu*.