



## Tentacule Mauve : La contre-attaque (TP2)

### Objectifs du travail

- Utiliser une interface graphique en Java avec Swing
- Thread
- Classe abstraite et/ou interface
- Écran de dialogues
- Réaliser un programme interactif
- Utiliser et traiter des événements
- MVC – Patron observateur
- Doit être fait en équipe de 2 personnes de même calibre, pré approuvé par l'enseignant

### Dates de remise

#### Le jeudi 8 décembre 2017 : Remise finale du projet sur Léa

- Le code source de votre programme
- Une autoévaluation de votre programme (ce qui fonctionne très bien, moins bien, pas du tout et pourquoi). Ce doit être un texte complet, même s'il est court.
- Les jeux d'essais utilisés pour tester votre programme, sous forme de tableau.

**NB.** Le travail complet doit être remis sur Léa pour **un seul** des deux membres de l'équipe.



## Mise en situation

Votre plan infallible n'a pas marché : Tentacule Mauve est de retour, et plus forte que jamais.

Comment est-ce possible ?

Votre jeu Boggle, lancé juste avant la version de Boggle de Tentacule Mauve (grâce à la machine à remonter dans le temps) aurait dû mettre un terme aux aspirations machiavéliques de Tentacule. Malheureusement, le bug de l'an 2000 avait lui-même un bug, et arriva 6515 jours plus tard que prévu. Faites un petit calcul rapide, ça tombe le 2 novembre 2017. Coïncidence ? Vous pouvez dire que je suis vieux jeu, mais je ne crois pas aux coïncidences.

L'économie s'effondra immédiatement. En moins de quelques minutes le taux de chômage est tombé à plus de 100%, et il y a eu des épidémies de sauterelles masquées un peu partout dans le monde en plus de la pire pénurie jamais observée de betteraves comestibles. Et, comme les mauvaises nouvelles n'arrivent jamais seules, votre cours de programmation a quand même lieu aujourd'hui à l'heure prévue. Que pouvait-il arriver de pire ?

Tentacule Mauve avait prévu le coup : elle a séquencé son ADN pour faire un, deux... un million de clones de Tentacule Mauve ! Afin de détruire l'humanité ! Mais vous n'avez pas dit votre dernier mot. Avec environ un mois devant vous, vous préparez la contre-attaque. Sans attendre, vous convoquez les gens les plus intelligents et les plus influents de la planète pour trouver une solution : des ingénieurs, des scientifiques, Obama, Angela Merkel, Justin Trudeau, Gandhi, Tony Stark, Shigeru Miyamoto et Ronald McDonald. L'équipe ultime. *The Dream Team*, comme on dirait chez ma grand-mère unilingue francophone (on dit tout le temps ça, s'en est rendu gênant).

Une décision fut rapidement trouvée (à l'unanimité moins un): vous allez personnellement attaquer cette invasion de Tentacules, sur un terrain défini d'avance.

Vous êtes notre dernier espoir. Mais comme nous n'avons pas autant confiance en vous qu'on le souhaiterait, on vous donne 3 chances. Comptez-vous triplement chanceux.



## L'attaque des clones

Le combat ultime : vous contre l'armée de Tentacule Mauve (ses clones et ses acolytes). Un champ de bataille. Le sort de l'humanité est entre vos mains. Mais aucune pression, faites votre possible 😊.



Fortement inspiré de : <https://www.youtube.com/watch?v=MtdyVHZluAw>

Mais bien lire l'énoncé comme il y a des différences.

## Précision concernant le jeu

### Vous (le héros)

Vous contrôlez le personnage principal, qui débute au centre de la joute de jeu. Vous pouvez vous déplacer de façon orthogonale, soit vers le haut, la droite, le bas et la gauche (avec les flèches du clavier). Vous tirez toujours dans la direction que vous regardez, à l'aide de la barre d'espace (*space*). Vous avez 3 points de vie, et après 3 collisions avec des ennemis vous êtes fatigués et vous vous couchez pour un long repos.

Arme de base : rayon laser qui se déplace plus vite que vous




### Image du héros

Vous utilisez toujours l'image qui correspond à la direction du héros, pour que le héros regarde dans la bonne direction (et qu'il tire dans la bonne direction).

## Ennemis




Les ennemis (E) se déplacent ainsi en fonction de la position du héros (H) :

### 3 exemples de déplacement des ennemis :

		
H	H	H
2 pixels en y, 0 en x	2 pixels en y, -2 en x où $ x  ==  y $	2 pixels en y, 2 en x où $ x  ==  y $

**Attention** : Les ennemis ne doivent jamais se superposer.

### 3 types d'ennemis à programmer :

Le méchant		Se déplace lentement, 1 point de vie	1 point
Le pas fin		Se déplace plus rapidement, 1 point de vie	2 points
Le trouble-fête		Se déplace lentement, 2 points de vie	3 points

### Les vagues d'ennemis

Les ennemis arrivent depuis l'un des 4 côtés de l'écran, par vague. Par exemple, 2 ennemis arrivent par le haut dans l'écran. Puis, 3 secondes plus tard, 2 autres ennemis arrivent par le bas dans l'écran. À chaque tranche de 3 secondes, de nouveaux ennemis apparaissent d'un nouveau côté.

Plus le temps avance, plus le jeu doit être difficile :

- Types d'ennemis plus difficile
- Plus d'ennemis d'un même côté
- Des ennemis arrivent simultanément de plusieurs côtés

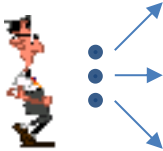
### Image des ennemis

Si les ennemis se déplacent vers le bas, la gauche ou la droite vous utilisez l'image « Front » de chacun des ennemis. Autrement, vous utilisez l'image « back » (ver le haut).

## Armes et bonus

Lorsque vous tuez une tentacule, il y a une possibilité qu'une capsule bonus tombe par terre à l'endroit où se trouvait l'ennemi. Si vous ramassez cette capsule, vous pouvez obtenir l'un des bonus suivant, choisi aléatoirement :

- Arme à balles : Tire trois balles dans la direction du joueur ainsi :



- Bombe zapper de tentacules : Élimine l'ensemble des ennemis dans l'écran
- Boost d'énergie : Permet de récupérer l'ensemble de ses points de vie

Chaque boni attrapé ajoute 5 points au total des points.

## Détection des collisions

Soyez logique dans la détection de vos collisions :

- Un de vos tirs de projectile doit détruire un ennemi, mais ne doit pas être en mesure de vous blesser.
- Une collision entre le héros et un ennemi élimine l'ennemi (peu importe le type d'ennemi), et vous perdez un point de vie.
- Il n'y a pas de collision entre les ennemis à gérer, comme ils ne doivent pas pouvoir se superposer.
- Ni vous ni les ennemis ne doivent pouvoir embarquer par-dessus une roche ou un buisson,
- Etc.

➤ Il faut utiliser les interfaces et/ou les classes abstraites dans la détection des collisions entre objets, et le déplacement des objets, pour rendre le programme plus facilement maintenable et évolutif.

## Fin de la partie

- Lorsque vous n'avez plus de points de vie (vous avez perdu)
- Autrement la partie est infinie, le but est d'obtenir le plus de points avant de mourir.

## Autres spécifications

### Pointage

Les points sont assignés en fonction du type d'ennemi et des bonus récupérés. Consultez les sections précédentes pour voir le nombre de points alloués pour chacun. Vous n'avez pas à sauvegarder les pointages, ni demander le nom du joueur.

### Points de vie

Le nombre de points de vie du héros doit être visible en tout temps, à l'aide de cœurs.

### Images

Toutes les images sont disponibles sur le R. Elles ne sont pas obligatoires, en ce sens que vous pouvez utiliser vos propres images si vous le désirez. Mais aucun temps ne sera alloué en classe pour travailler sur vos images, ce devra être fait à l'extérieur des plages horaires de classe.

## Autres directives

- MVC + patron observateur uniquement pour :
  - le pointage
  - les points de vie
  - Détection de fin de jeu
- Vous devez afficher en tout temps les informations pertinentes au jeu.
- Il doit y avoir un menu :
  - Fichier
    - Nouvelle partie
    - -----
    - Quitter
  - ?
    - Aide (Affiche un JDialogBox avec les règles sommaires)
    - -----
    - À propos (Vos noms, date de création du projet final)

### De plus :

- Vous serez noté sur les classes créées, leurs hiérarchies, la portée des variables et méthodes etc. Pensez-y plus tôt plutôt que plus tard.
  - Respecter les normes de programmation, et les standards de programmation orienté objet (ex : l'encapsulation des données, la surcharge des constructeurs, etc).
  - Vous serez évalué sur l'esthétique et la convivialité de l'application.
-

## Critères d'évaluation

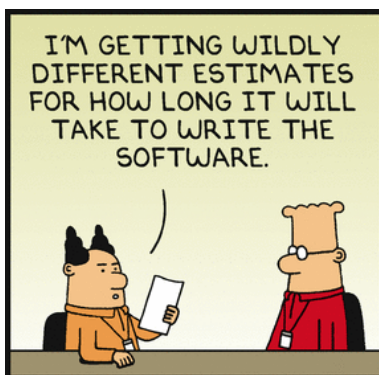
Votre programme doit respecter les normes de programmation.

Vous devez réaliser ce travail en équipe de 2, sauf exception.

Assurez-vous que votre travail compile sous l'environnement NetBeans 8 des laboratoires du Collège.

Critère	Nombre de points accordés
Document d'accompagnement du programme Jeux d'essai Auto-évaluation	/2
Esthétique et convivialité <ul style="list-style-type: none"> <li>▪ Groupement logique de composants</li> <li>▪ Homogénéité entre fenêtres</li> <li>▪ Sélection judicieuse des composants utilisés</li> <li>▪ Originalité/Complexité/esthétique des composants graphiques</li> </ul>	/3
Programmation <ul style="list-style-type: none"> <li>▪ Clarté, simplicité (utilisation judicieuse des instructions de Java)</li> <li>▪ Respect des normes de programmation orientée objet (Classes, hiérarchie, encapsulation etc.)</li> <li>▪ MVC + Patron observateur</li> <li>▪ Thread</li> <li>▪ Javadoc</li> </ul>	/4
Fonctionnement du programme <ul style="list-style-type: none"> <li>▪ Résultats obtenus</li> <li>▪ Respect des consignes</li> </ul>	/11
<b>Total</b>	<b>/20</b>

### Minute de blague gratuite





## Annexe

### Code pour gérer plusieurs touches appuyées simultanément

```
private ArrayList<Integer> listeKeyCodes = new ArrayList<>();

private void creerEvenements() {
    addKeyListener(new KeyAdapter() {

        @Override
        public void keyPressed(KeyEvent e) {
            if (!listeKeyCodes.contains(e.getKeyCode())) {
                listeKeyCodes.add(e.getKeyCode());
            }
        }

        @Override
        public void keyReleased(KeyEvent e) {
            // On enlève l'objet Integer, on ne veut pas
            // enlever une position dans l'ArrayList
            listeKeyCodes.remove(new Integer(e.getKeyCode()));
        }

    });
}

// pour valider si la touche 'A' est présente dans
// l'ArrayList :
if (listeKeyCodes.contains(KeyEvent.VK_A)) {
    // Code pour gérer la touche 'A'
}
```

ps. C'est à vous de trouver comment communiquer cette information au jeu ! ☺

### Minute de blague bonus





## Boni supplémentaire

*Construire son propre Tentacule Mauve!*

