

Travail pratique #1

Graphiques en 2D

INF5071 - Infographie

Automne 2023

Table des matières

Exercice 1 : Respiration profonde (8 pts)	1
Partie 1 : Préparation de l'animation	2
Partie 2 : SVG et animation	2
Exercice 2 : Modélisation 2.5D (12 pts)	3
Partie 1 : Ciel étoilé & Environnement	3
Partie 2 : Effet parallaxe	4
Partie 3 : Modélisation d'un vaisseau	4
Partie 4 : Interactivité avec le clavier	4
Instructions	5
Références	5

Exercice 1 : Respiration profonde (8 pts)

Pour cet exercice, vous devez utiliser Krita, Inkscape et SVG

La respiration profonde est une technique souvent utilisée pour calmer l'anxiété. Elle consiste à suivre un cycle lent d'inspirations et d'expirations, entrecoupées de pauses. Par exemple, on peut

1. Inspirer pendant 5 secondes,
2. Attendre 5 secondes,
3. Expirer pendant 5 secondes,
4. Attendre 5 secondes,

Le but de l'exercice 1 est de créer une animation SVG pour guider la respiration profonde. Le résultat final de cet exercice devrait être similaire à la figure suivante (voir la version animée sur Moodle).

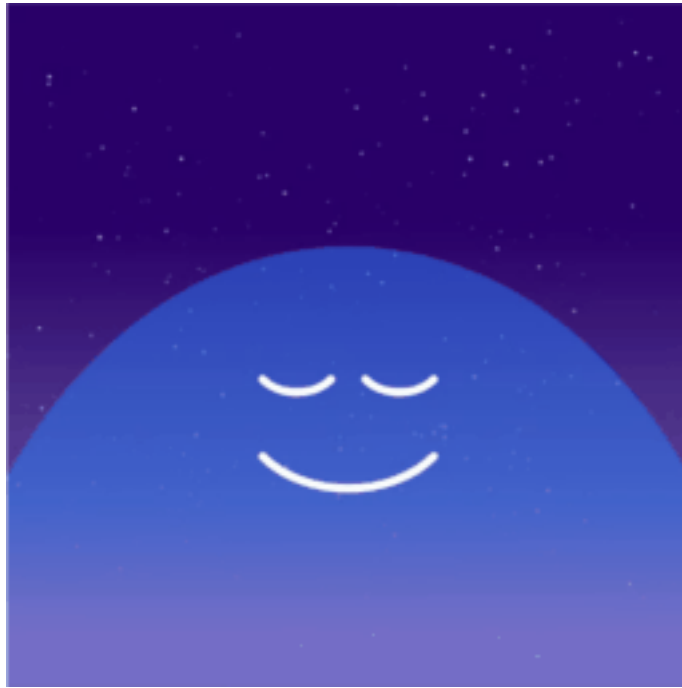


Figure 1: Exemple de résultat pour l'exercice 1

Partie 1 : Préparation de l'animation

- Utilisez le logiciel de peinture Krita pour créer une image jpg qui sera utilisée pour l'arrière-plan de votre animation finale. L'arrière-plan carré doit être formé d'un gradient de couleur, et vous devez ajouter des étoiles. Fournissez votre fichier krita (`tp1_ex1_ciel.kra`) et l'image générée (`tp1_ex1_ciel.jpg`).
- Utilisez le logiciel de dessin Inkscape pour créer un croquis du personnage que vous dessinerez et animerez à la partie 2. Utilisez des primitives géométriques simples pour former votre dessin. Exportez votre croquis en format `.png`. Fournissez votre fichier Inkscape (`tp1_ex1_prepa.svg`) et votre image png (`tp1_ex1_prepa.png`).

Partie 2 : SVG et animation

- Programmez votre dessin (fichier `tp1-ex1.svg`) dans un éditeur texte ou un IDE. Utilisez votre croquis préparé à la partie 1 pour guider la programmation de votre personnage.
- Ajoutez ensuite une animation de respiration. Vous devez animer : (1) la position du visage (bouche et yeux) et (2) la hauteur de la tête. Votre animation doit représenter les 4 phases du cycle de respiration profonde (inspiration, attente, expiration, attente). Chaque phase doit être d'une durée de 5 secondes.
- Ajouter du texte animé pour représenter la phase d'inspiration / d'expiration. L'opacité du texte doit être animée.

Exercice 2 : Modélisation 2.5D (12 pts)

Pour cet exercice, vous devez utiliser HTML Canvas pour développer un jeu 2D simple.

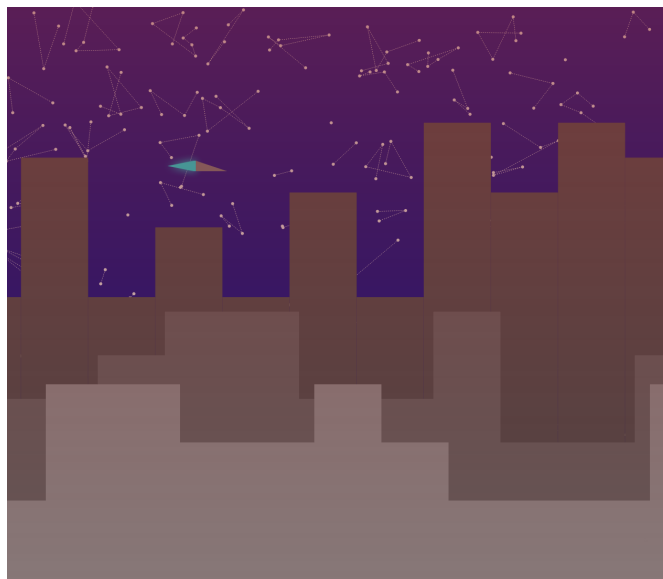


Figure 2: Exemple de résultat pour l'exercice 2

Utilisez le fichier `tp1_ex2.html` pour compléter cet exercice. Nous utilisons ici HTML Canvas pour les graphiques, JavaScript pour contrôler notre boucle d'animation et pour l'interactivité avec le clavier. La vidéo suivante (<https://youtu.be/y9iZqcwAiLo>) vous montre le genre de résultat attendu pour cet exercice.

Partie 1 : Ciel étoilé & Environnement

1. Au lieu de créer le ciel avec un logiciel de peinture, il sera généré de façon procédurale pour cet exercice. Pour cette partie, vous devez modifier la fonction `draw_background()` afin de dessiner le ciel étoilé. Utilisez les données générées par la fonction `generate_constellations()` qui est exécutée lors de l'initialisation de l'application graphique. La variable `constellations` est une liste de constellation, chacune contenant une liste d'étoiles et de leur position (x, y) entre 0 et 1. Le ciel doit être formé d'une couleur sombre uniforme, les étoiles doivent être représentées par des cercles de `rayon=3px`, et le lien entre les étoiles doit être dessiné avec des lignes pointillées d'épaisseur `w=2px` afin de représenter les constellations.
2. Afin de créer l'environnement de votre jeu, nous allons utiliser un gradient linéaire semi-transparent pour simuler le smog. Modifiez la routine `draw_smog` pour réaliser cet effet. Le gradient linéaire doit faire varier la couleur verticalement. Cette fonction est la dernière exécutée dans la routine `draw()` et modifiera donc la teinte de tous les dessins qui seront dessinés avant l'application du gradient.

Partie 2 : Effet parallaxe

Pour simuler la profondeur, nous utiliserons l'effet parallaxe. Cet effet peut être créé en générant de façon procédurale des bâtiments rectangulaires placés à 3 différentes profondeurs. Pour simuler la profondeur, chaque rangée se déplace à une vitesse de plus en plus lente (p. ex. les bâtiments les plus rapprochés se déplacent rapidement, les bâtiments les plus éloignés sont plus lents.)

Pour cette question, générez aléatoirement une liste de 64 bâtiments ayant chacune une hauteur aléatoire. Utilisez `Math.random` pour la génération des nombres aléatoires. Utilisez des rectangles de trois couleurs différentes pour représenter la profondeur. Lorsque les bâtiments ne sont plus dans le canevas, leur position doit être ajustée pour qu'ils se retrouvent au bout de la liste à droite du canevas, ce qui permet de simuler une ville de longueur infinie avec un nombre limité de bâtiments.

Partie 3 : Modélisation d'un vaisseau

Modélisez un vaisseau spatial qui sera dessiné par-dessus la ville en mouvement au centre du canevas. Utilisez les formes de votre choix. Pour représenter une propulsion à plasma, utilisez les styles `shadowBlur` et `shadowColor` de `HTML Canvas`. Modifiez les routines `draw_vessel`.

Partie 4 : Interactivité avec le clavier

Ajoutez une interactivité minimale pour ce squelette de jeu. Voici les fonctionnalités désirées :

- Touche A et Flèche vers la gauche : Déplace le vaisseau vers la gauche, sans dépasser la limite du canevas
- Touche S et Flèche vers le bas : Déplace le vaisseau vers le bas, sans dépasser la limite du canevas
- Touche D et Flèche vers la droite : Déplace le vaisseau vers la droite, sans dépasser la limite du canevas
- Touche W et Flèche vers le haut : Déplace le vaisseau vers le haut, sans dépasser la limite du canevas
- Touche Espace : Augmente la vitesse du vaisseau et allonge sa plume de propulsion (partie bleue dans la figure 2)
- **Note** : Les codes de clé (*keycode*) pour les touches du clavier sont disponibles sur le site suivant : <https://tutorial.eyehunts.com/js/javascript-keycode-list-event-which-event-key-event-code-values/>

Pour modifier la vitesse du vaisseau, utilisez l'équation suivante :

$$v_{k+1} = \begin{cases} v_{min} & \text{si la touche espace n'est pas enfoncée} \\ v_k + a & \text{si la touche espace est enfoncée et que } v_k < v_{max} \\ v_{max} & \text{sinon} \end{cases} \quad (1)$$

où v_k est la vitesse du vaisseau au frame k , v_{min} et v_{max} sont les vitesses minimale et maximale permises, et a est l'accélération du vaisseau.

Modifiez les routines `updateVesselPosition` et `releaseSpace`, utilisez la variable `vesselPos` et utilisez une transformation de type translation dans la méthode `draw_vessel` pour le placer au

bon endroit avant de le dessiner.

Instructions

- Le travail pratique doit être réalisé en équipe de 2 maximum.
- Vous devez être membre d'un groupe pour avoir accès à la remise du TP. Vous devez choisir votre groupe (seul ou à plusieurs) grâce à l'activité "choix d'une équipe pour les TPS" sur Moodle.
- Vous devez soumettre un seul fichier **zip** qui contient tous les fichiers nécessaires pour tester votre travail.
- Votre fichier zip doit être nommé comme suit :

`tp1_<code_MS_Etudiant1>_<code_MS_Etudiant2>.zip`

- Le fichier compressé doit contenir les fichiers suivants
 - `tp1_ex1_ciel.kra`
 - `tp1_ex1_ciel.jpg`
 - `tp1_ex1_prepa.svg`
 - `tp1_ex1_prepa.png`
 - `tp1_ex1.svg`
 - `tp1_ex2.html`
- Veuillez utiliser les gabarits fournis avec le TP pour commencer les exercices.
- Vous devez obligatoirement soumettre votre travail via Moodle . Les soumissions par courriel ne seront pas corrigées.
- Le plagiat ne sera pas toléré, écrivez votre propre code. Les normes de plagiat de l'université seront appliquées en cas de plagiat (voir la Politique 18).
- Il est important qu'il n'y ait pas d'erreurs d'exécution pour la correction. Les images doivent s'afficher.
- Indiquez en commentaire dans le fichier `tp1-ex2.html` quel fureteur web vous avez utilisé pour réaliser le TP. Veuillez utiliser Firefox, Chrome ou Edge.
- Une partie des points pour chaque question est attribuée à la lisibilité du code. Mettre des commentaires pour expliquer votre démarche.
- La qualité artistique n'est pas un objectif du cours. Ne passez donc pas trop de temps sur l'esthétique de vos graphiques, puisque cela ne sera pas évalué.
- **Note** : ce travail compte pour 15% de la note finale.

Références

- https://www.w3schools.com/colors/colors_picker.asp
- https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API
- <https://developer.mozilla.org/en-US/docs/Web/SVG>