

Projet 1 : Bubble struggle

Présentation - objectif

Bubble struggle est un jeu video dont la première version date des années 90. Dans ce jeu, vous incarnez un personnage attaqué par des bulles. Le personnage doit détruire des bulles à l'aide d'un harpon pour atteindre le niveau suivant. A chaque fois qu'une bulle est atteinte, soit elle est suffisamment petite et elle disparaît, soit elle ne l'est pas et se divise en deux bulles plus petites qui partent chacune d'un côté de l'écran.



Figure 1: Image d'une version en ligne actuelle du jeu en html5 ([lien vers le jeu](#))

Votre objectif dans ce projet est de reproduire les mouvements des bulles, sans les animer : vous devrez obtenir des affichages du type :

Vous devrez créer au moins une classe qui assurera l'essentiel du fonctionnement du projet.

La rencontre avec un harpon en cours de mouvement devra pouvoir être simulée (la bulle se divise en deux et chacune des bulles résultantes se dirige d'un côté).

Outils - Ressources

Affichage

L'affichage sera délégué à la bibliothèque Tkinter de python.

Comme dans de nombreux cas en traitement de l'image, Tkinter place l'origine

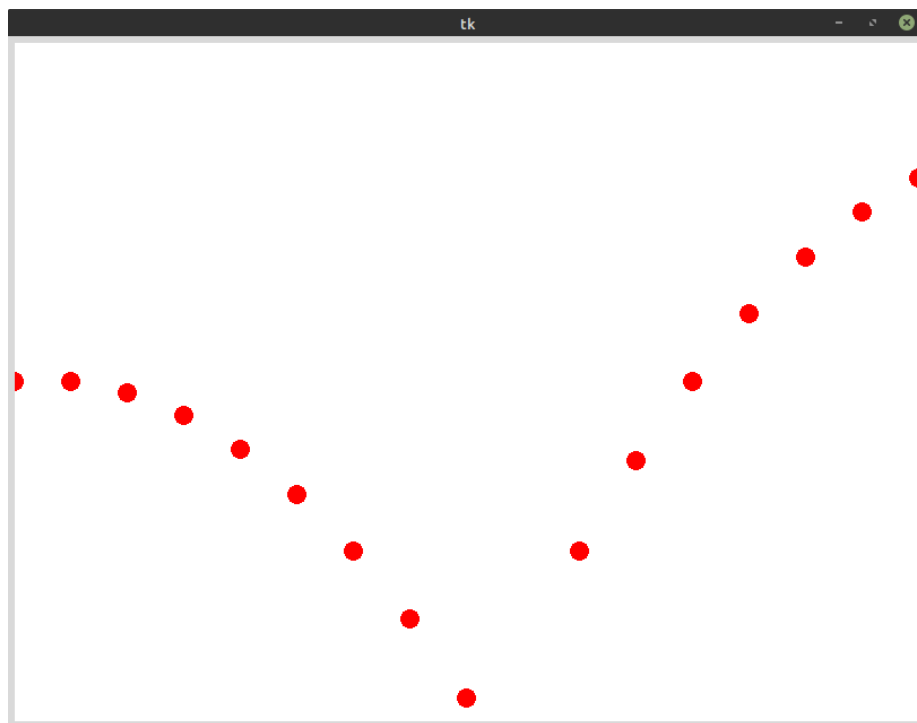


Figure 2: Exemple d’affichage obtenu dans votre projet : une seule balle part de la droite avec une vitesse vers la gauche et vers le bas

du repère en haut et à gauche de la fenêtre :

- le point de coordonnées (0,0) est en haut à gauche,
- le point de coordonnées (0,hauteur) où hauteur est la hauteur de la fenêtre d'affichage est en bas et à gauche de celle-ci,
- le point de coordonnées (largeur,0) où largeur est la largeur de la fenêtre d'affichage est en haut et à droite de celle-ci,
- le point de coordonnées (largeur,hauteur) est en bas à droite de la fenêtre.

Un exemple de code commenté utilisant cette bibliothèque pour afficher un cercle sur fond blanc :

```
from tkinter import *

# Constantes pour l'affichage : dimensions de la fenêtre
Largeur = 800
Hauteur = 600

# Fenêtre tkinter
from tkinter import *
root = Tk()
canvas=Canvas(root,width=Largeur,height=Hauteur,background="white")
canvas.pack(side=LEFT, padx=5, pady=5)

r=30 # rayon du cercle
x,y = (300,200) # coordonnées du centre du cercle
canvas.create_oval(x-r,y-r,x+r,y+r,width=1, outline="red",fill='blue')
# width fixe l'épaisseur du trait
# outline donne la couleur de celui-ci
# fill est la couleur de remplissage

root.mainloop()
```

Obtenir un comportement physique réaliste des bulles

Dans le jeu, les bulles se comportent comme de gros ballons :

- elles tombent,
- rebondissent au sol et contre les murs et
- ralentissent à cause du frottement de l'air.

Un modèle physique simplifié de ces phénomènes peut être (D'après le très bon site nsi-a-feuillade.fr):

Modélisation

Une particule est modélisée par cinq valeurs :

- ses coordonnées x et y ,

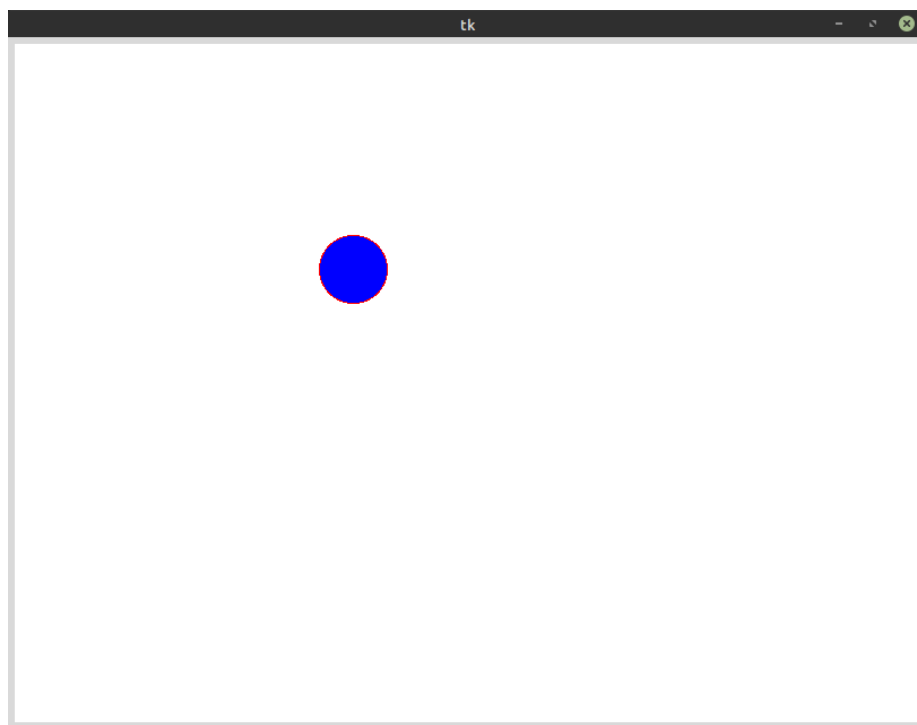


Figure 3: Exemple d’affichage d’un cercle avec Tkinter

- les coordonnées vx et vy de sa vitesse,
- sa masse m .

Particule soumise à aucune force

Si aucune force n'agit sur la particule, alors elle conserve sa direction et sa vitesse. Ainsi à l'instant élémentaire suivant, la nouvelle position de la particule est :

$$\begin{aligned}x &= x + vx * t \\ y &= y + vy * t\end{aligned}$$

t est la durée entre deux positions calculées de la particule. Comme vous devez ici seulement afficher les positions successives, vous pourrez choisir $t=1$.

La particule a donc un mouvement rectiligne uniforme.

Particule soumise à son poids

Si la particule est soumise à son poids alors à l'instant suivant, la nouvelle vitesse de la particule est :

$$\begin{aligned}Vx &= Vx \\ Vy &= Vy - g * t\end{aligned}$$

Où g est l'accélération de la gravité. g ici vaut 10.

Comme la vitesse est modifiée, cela induit un changement sur la future position.

Particule soumise aux frottements (facultatif)

On note $V = (Vx^2 + Vy^2)^{0.5}$. Les nouvelles valeurs du vecteur vitesse sont données par les formules :

- $V'_x = Vx - (f/m) * (V^e) * (Vx / V)$
- $V'_y = Vy - (f/m) * (V^e) * (Vy / V)$

où f est le coefficient de frottement et e l'exposant de frottement.

Essayer différentes valeurs du coefficient de frottement ($f=0.005$ peut être un bon point de départ) et surtout de l'exposant. Un exposant $e = 1$ correspond à un frottement pour une bulle se déplaçant à faible vitesse ; un exposant $e = 2$.

Interaction avec le côtés et le sol

La balle doit rebondir au sol et contre les murs de droite et de gauche, mais pas au plafond. Pour cela, l'un des composantes de la vitesse doit être inversée, et le mouvement doit être modifié au moment du rebond.

Déroulement du projet

Le projet se déroule par groupe de 2 ou 3. Le découpage présenté ici s'appuie sur le travail de stage proposé par Mme Delphine Taranowicz a propos des tours de Hanoï.

Travail du fonctionnel - en classe - le 13 Octobre

Réécrire avec vos propres mots, en groupe ce que vous attendez que réalise votre projet. Une autre façon de décrire cette étape : décomposer ce que vous devez réaliser en grandes étapes qui permettent de résoudre le problème.

Prévoir des tests pour chaque étape, afin que les étapes à venir soient déjà guidées.

Exemple sur un problème déjà vu : le jeu de bataille :

- créer le jeu de cartes,
- le mélanger,
- partager le jeu en deux,
- gérer un tour de jeu,
- gérer une partie.

Test possible pour cet exemple :

- vérifier le nombre des cartes du jeu
- vérifier que les deux premières et les deux dernières cartes ne se suivent pas
- prévoir l'état des 2 jeux avant un tour et après ce tour
- prévoir une ou des partie(s) test avec l'état du jeu avant et après la partie.
- prévoir de pouvoir tester la fin du jeu, l'égalité...

Vous rendrez un document papier par groupe après une heure de travail.

Travail de l'analyste - en classe - le 18 Octobre

Décomposer votre projet en classes(s)) dont vous spécifierez les attributs et méthodes. Vous devrez vous appuyer sur le travail du fonctionnel.

Exemple sur un problème déjà vu : la pile :

- la classe `cellule` qui a deux attributs : `valeur` et `suivant`.
- la classe `pile` qui a un attribut : le `sommet` et, notamment les méthodes `est_vide`, `empiler`, `dépiler`... dont on doit décrire les préconditions et postconditions.

Vous rendrez un document papier par groupe après une heure de travail.

Travail du programmeur - début en classe le 22 Octobre, travail à poursuivre et à rendre par pronote au plus tard le 15/11/21

Implanter votre code afin de réaliser le projet.

Vous documenterez chaque classe et méthode à l'aide de sa docstring.

Vous prévoirez un fichier de tests et en ferez usage pour valider votre code.

Un point d'étape sur cette partie est prévu le 8 Novembre. Vous devez prévoir de présenter votre travail au professeur à cette date.

Travail de présentation à rendre au plus tard le 19/11 en suivant le lien pronote dédié

Votre vidéo, de 180 secondes, présentera un point particulier de votre projet : pourquoi ce point vous a semblé intéressant, les difficultés éventuelles que vous y avez rencontré et éventuellement d'autres solutions avec leurs avantages et inconvénients. Toutes les étapes de ce projet (fonctionnel, analyste, programmeur...,) peuvent être l'objet de votre présentation.

Evaluation

Les compétences évaluées ici seront :

Compétence	fonctionnel	analyste	programmeur	présentation
analyser et modéliser un problème en termes de flux et de traitement d'informations	*			
décomposer un problème en sous-problèmes, reconnaître des situations déjà analysées et réutiliser des solutions	*	*	*	

Compétence	fonctionnel	analyste	programmeur	présentation
concevoir des solutions algorithmiques	*			
traduire un algorithme dans un langage de program- mation, en spécifier les interfaces et les inter- actions, compre- dre et réutiliser des codes sources existants, développer des processus de mise au point et de validation de programmes			*	
développer des capacités d'abstraction et de généralisation		*		*

Compétence	fonctionnel	analyste	programmeur	présentation
Communiquer à l'écrit comme à l'oral : présenter une démarche de manière argumen- tée, synthé- tique et cohérente ; utiliser un vocabu- laire adapté et choisir des modes de représenta- tion appropriés; échanger entre pairs.	*	*		*