

Programmer un jeu vidéo avec Pyxel : 5/6

Ajouter des explosions lors des collisions

Nous repartirons dans ce tutoriel du script réalisé précédemment : *tuto_pyxel_4.py*

Faire Enregistrer Sous pour renommer à présent ce script en *tuto_pyxel_5.py*

1. Pour démarrer...

Rappels vus lors du tutoriel précédent :

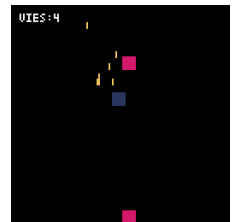
- Quelle était la forme de la zone de détection d'une collision ?
- Comment supprimer un élément *x* dans une liste ?
- Comment écrire « GAME OVER » sur la fenêtre de jeu ?

On voudrait maintenant que la destruction d'un ennemi soit matérialisée par une explosion, sous forme d'un cercle qui grandit progressivement puis disparaît (comme une onde de choc concentrique)

Pour cela, on va créer deux nouvelles fonctions : ***explosions_creation(x,y)*** et ***explosions_animation()*** qui seront toutes les deux appelées par d'autres fonctions du jeu.

2. Créer une explosion

La fonction ***explosions_creation(x,y)*** va s'occuper de créer un cercle à l'emplacement de la collision. A quel(s) endroit(s) dans le code devra-t-on utiliser cette fonction, et avec quels paramètres ?



.....
.....

Chaque explosion sera représentée en mémoire par une liste $[x, y, d]$:

- les deux premiers paramètres sont les coordonnées désignant le vaisseau ou l'ennemi qui est touché
- le troisième paramètre représente le diamètre du cercle d'explosion. On partira d'un diamètre égal à zéro (il augmentera ensuite lors de l'animation).

Les explosions sont stockées dans une liste : *explosions_liste* : c'est une liste vide au départ, et la fonction ***explosions_creation(x,y)*** se contente d'y ajouter la nouvelle explosion créée.

Jalon 1 : écrire le code de cette fonction, ainsi que les deux appels d'utilisation, sans oublier l'initialisation de la liste des explosions.

3. Animer une explosion

Voici le code de la fonction ***explosions_animation()***

```
def explosions_animation():  
    """animation des explosions"""  
    for explosion in explosions_liste:  
        explosion[2] +=1  
        if explosion[2] == 12:  
            explosions_liste.remove(explosion)
```

Expliquer le mécanisme d'animation des explosions :
.....
.....
.....

Où sera appelée cette fonction ? Intégrer l'appel au bon endroit dans le code.

4. Dessiner concrètement l'explosion

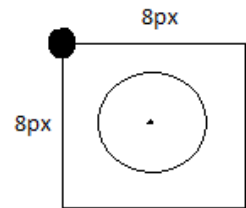
Pour l'instant, les explosions et leur animation sont représentées dans la mémoire de l'ordinateur, mais on ne les a pas encore dessinées à l'écran !

Où va-t-on écrire les instructions pour ce faire ?

Voici un extrait de la documentation de Pyxel, concernant la méthode **circb**

```
circb(x, y, r, col)
```

Dessine le contour d'un cercle de rayon **r** et de couleur **col** à (**x**, **y**)



On veut que le cercle soit bien centré sur l'explosion. Quelles doivent être les coordonnées du centre du cercle ?

Si le cercle est de couleur grise (code couleur 13), écrire alors l'instruction qui dessine le cercle au bon endroit

On voudrait pour faire plus joli que, lors de l'animation, les cercles changent de couleur. Pour cela, on met le dernier paramètre à $8 + explosion[2]\%3$

0	#000000 0, 0, 0	1	#2B335F 43, 51, 95	2	#7E2072 126, 32, 114	3	#19959C 25, 149, 156
4	#BB4B52 139, 72, 82	5	#395C9B 57, 92, 152	6	#A9C1FF 169, 193, 255	7	#EEEEEE 238, 238, 238
8	#D41B6C 212, 24, 108	9	#D3B441 211, 132, 65	10	#E9C35B 233, 195, 91	11	#70C6A9 112, 198, 169
12	#7696DE 118, 150, 222	13	#A3A3A3 163, 163, 163	14	#FF979B 255, 151, 152	15	#EDC780 237, 199, 176

Compléter le tableau ci-dessous avec les différentes valeurs prises successivement par ce paramètre

Diamètre									
Indice couleur									
couleur									

Jalon final : animation des explosions finalisées