

Il s'agit de réaliser un jeu qui simule, de façon très simplifiée, un tennis de table. Chaque joueur commande une raquette et la balle peut rebondir sur les bords haut et bas de l'écran. Le premier joueur qui atteint 7 points a gagné la manche.

### 1 Préparation de l'espace de travail

Pour cette activité, nous n'utiliserons pas le lutin fourni par défaut. Nous pouvons donc le supprimer (voir ci-contre).

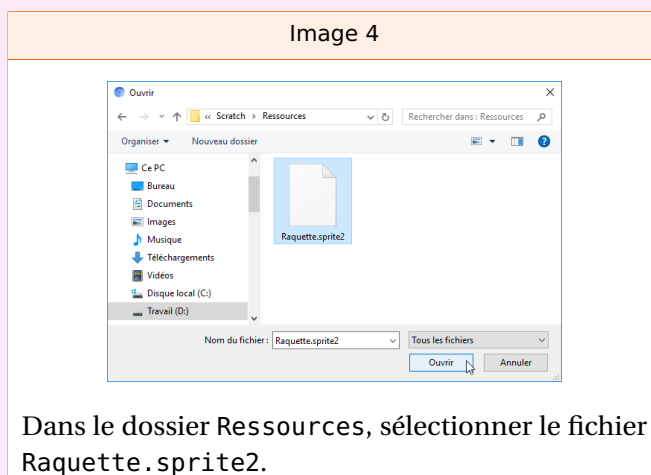
En revanche, nous aurons besoin de deux lutins différents pour représenter les raquettes et la balle :

- en suivant la manipulation Pour importer un lutin, décrite ci-après, importer (une seule fois) le lutin *Raquette*;
- en suivant la manipulation Pour choisir un lutin dans la bibliothèque, décrite ci-après, créer un nouveau lutin *Ball*.

#### Pour supprimer un lutin :



#### Pour importer un lutin :



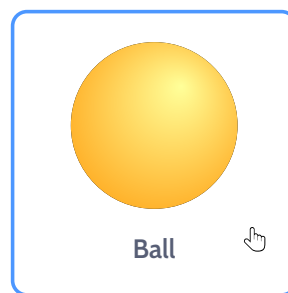
### Pour choisir un lutin dans la bibliothèque :

Image 5



Cliquer sur le bouton  (choisir un sprite).

Image 6



Sélectionner le lutin *Ball*.

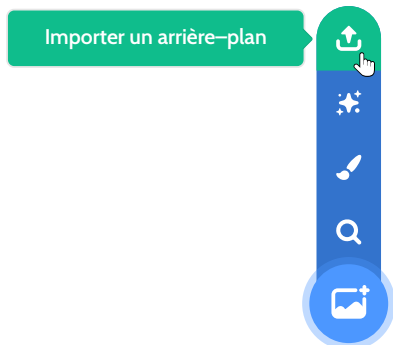
À l'issue de cette étape, on pourra :

- renommer le lutin *Raquette* en *Raquette\_gauche*;
- supprimer tous les costumes, sauf le premier, du lutin *Ball*;
- supprimer l'arrière-plan à fond blanc;
- remplir d'une couleur uniforme le lutin *Ball*.

À l'aide des explications ci-dessous, importer l'arrière plan *pong . png*.

### Pour importer un nouvel arrière-plan :

Image 7




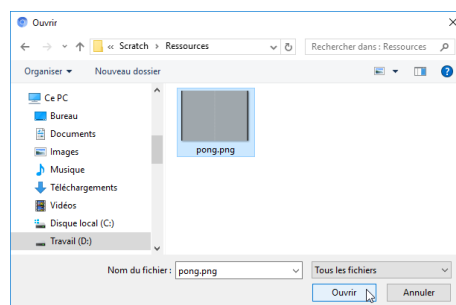
Cliquer sur le bouton  (Importer un arrière-plan).

Image 8



Dans le dossier *Ressources*, sélectionner le fichier *pong . png*.

## 2

## Initialisation du programme

Plusieurs événements, définis à l'aide de différents messages, permettront de gérer l'activité des lutins :

- le message *Nouvelle partie* permet de replacer les lutins dans leur état initial et d'initialiser les variables à leurs valeurs initiales;
- le message *Nouvelle balle* replace la balle au centre de la scène, en réinitialisant sa vitesse, et les raquettes en leur position initiale;
- le message *Vitesse balle* permet d'augmenter progressivement la vitesse de la balle;
- le message *Fin de partie* permet d'afficher le nom du vainqueur et lance l'attente d'une nouvelle partie.

quand je reçois

Fin de partie ▼

Nouveau message

✓ Fin de partie


Nouvelle balle

Nouvelle partie

Vitesse balle

Pour ce jeu, nous utiliserons les variables suivantes :

- **Score\_Joueur\_1** et **Score\_Joueur\_2** (initialement à 0) pour conserver le score actuel de chaque joueur;
- **VBalle** (initialement à 10) qui caractérise la vitesse de la balle;
- **Vraquettes** (par défaut à 10) qui caractérise la vitesse des raquettes.

Dans Scratch, créer les variables précédemment décrites puis sélectionner la scène et, dans l'onglet  **Code**, inscrire le script d'initialisation suivant :

Scène



Arrière-plans  
1

Programme 1

**Quand  est cliqué**

envoyer à tous **Nouvelle partie** ▾

**quand je reçois **Fin de partie** ▾**

mettre l'effet **fantôme** ▾ à **50**

**quand je reçois **Nouvelle partie** ▾**

annuler les effets graphiques

mettre **Vraquettes** ▾ à **10**

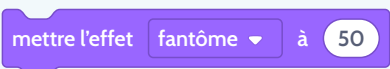
mettre **VBalle** ▾ à **10**

mettre **Score\_Joueur\_1** ▾ à **0**

mettre **Score\_Joueur\_2** ▾ à **0**

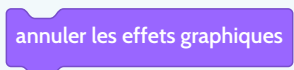
envoyer à tous **Nouvelle balle** ▾

Aide



Ce bloc permet d'ajouter un effet graphique (à choisir dans la liste déroulante) à un lutin. Certains effets vont de 0 à 100 tandis que d'autres peuvent aller de -100 à 100.

Aide




Ce bloc permet de supprimer les effets graphiques attribués à un lutin.

Pour modifier l'affichage d'une variable :

Image 9

**Score\_Joueur\_1** 0

lecture normale

grande lecture 


barre de défilement

Effectuer un clic avec le bouton droit de la souris sur une variable et sélectionner le menu grande lecture .

On cachera l'affichage des variables **VBalle** et **Vraquettes** tandis qu'on choisira un affichage grande lecture pour les variables **Score\_Joueur\_1** et **Score\_Joueur\_2** .

### 3 Les raquettes

La raquette de gauche a pour abscisse -225 et son déplacement vertical est obtenu à l'appui de la touche **Z** (vers le haut) ou de la touche **W** vers le bas.

Déplacer la raquette consiste donc à modifier son ordonnée y en lui ajoutant ou en lui retirant la valeur définie par la variable **Vraquettes** . Pour ce faire, on utilisera le bloc  .

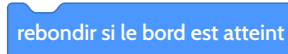


En mathématiques, soustraire consiste à ajouter l'opposé. Dans Scratch, comme il n'existe pas d'instruction **enlever** à *y*, on utilise cette propriété pour retrancher **Vraquettes** à *y*.



Écrire le script associé au lutin *Raquette\_gauche* qui permet de le déplacer verticalement. Le script doit se déclencher quand l'événement *Nouvelle balle* est reçu.

On pourra utiliser, en particulier, les blocs suivants :



Répéter indéfiniment :

Si la touche Z est pressée alors :

Ajouter **Vraquettes** à *y*

Fin du test

Si la touche W est pressée alors :

Retirer **Vraquettes** à *y*

Fin du test

Rebondir si le bord est atteint

Fin de la boucle

Algorithme

Aide

rebondir si le bord est atteint

Lorsqu'il est utilisé conjointement avec le bloc **avancer de** pas, ce bloc permet au lutin de rebondir contre l'un des bords de la scène, d'un angle égal à celui qu'il faisait avec le bord avant de toucher celui-ci.

Lorsque ce bloc est utilisé avec une autre instruction, telle que **ajouter** à *y*, il permet d'éviter que le lutin ne sorte de la scène.

À l'issue de cette étape, et après avoir vérifié le bon fonctionnement du script :

- dupliquer le lutin *Raquette\_gauche*;
- renommer le nouveau lutin *Raquette\_droite*;
- modifier les éléments qui doivent l'être dans le script associé au lutin *Raquette\_droite*.

## 4 La balle

Quand l'événement *Nouvelle balle* est reçu, les tâches à accomplir par le lutin *Ball* peuvent être décomposées de la manière suivante :

### Initialisation

Affichage du lutin  
Placement au centre de l'écran  
Orientation aléatoire  
Vitesse de la balle par défaut

### Rebond

La balle rebondit tant qu'elle ne touche pas la couleur noire  
La vitesse augmente progressivement


### Scores

Mise à jour des scores  
Test si fin de partie atteinte

## 1 Initialisation

Ajuster la taille du lutin  
Placer le lutin au centre de l'écran  
Orienter à nombre aléatoire entre 20 et 130  
Montrer le lutin  
Attendre 1 seconde  
Envoi du message *Vitesse balle*

Algorithme

Par défaut, la taille du lutin *Ball* est trop grande. On commencera donc par ajuster sa taille à l'aide de l'instruction  (35 % paraît un bon compromis).


Les actions à effectuer à réception du message *Vitesse balle* seront traitées ultérieurement.

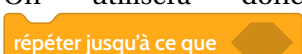
Aide

attendre 2 secondes

Ce bloc permet d'interrompre un script pendant la durée passée en paramètre avant de reprendre l'exécution du script au bloc suivant.

## 2 Rebond

Le mouvement de la balle est relativement aisé à appréhender : la balle avance continuellement de , en suivant l'orientation fixée, tant qu'elle n'atteint pas une zone noire de la scène.


On utilisera donc une boucle du type  pour y inclure les instructions




destinées à gérer le mouvement de la balle.

Répéter jusqu'à couleur noire touchée :

Si la raquette est touchée :

S'orienter à - 

Tourner à droite  de nombre aléatoire entre -5 et 5

Fin du test

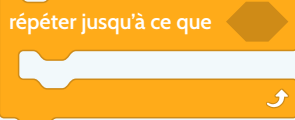
Rebondir si le bord est atteint

Avancer de 

Fin de la boucle

Algorithme

Aide




Ce bloc prend en paramètre une condition : tant que la condition n'est pas vérifiée, les instructions situées à l'intérieur du bloc sont répétées.

Pour détecter le moment où la balle touche une raquette ou sort de la zone de jeu, il suffit de déterminer si le lutin *Ball* entre en contact avec la couleur noire ou avec la couleur choisie pour la raquette.


Aide



Ce bloc (que l'on utilise à l'intérieur d'un test) permet de détecter lorsqu'un lutin touche une couleur donnée.

Pour sélectionner une couleur, il faut d'abord cliquer sur l'ovale coloré puis choisir une couleur ou bien se servir de la pipette  et cliquer sur un élément de la couleur désirée.

Au cours de son mouvement, si la balle touche le bord inférieur ou supérieur de la scène, ou si elle touche une raquette, elle rebondit :

- le rebond sur le bord inférieur ou supérieur de la scène s'effectue à l'aide de  ;
- pour effectuer le rebond sur l'une des raquettes, il suffit de modifier l'orientation de la balle en lui faisant prendre la direction opposée à celle de sa course avant de toucher la raquette.

Aide

direction

Ce bloc renvoie la mesure de l'angle, en degrés, selon laquelle le lutin est orienté.

Au moment du rebond sur une raquette, on modifiera légèrement (et aléatoirement) l'orientation du lutin de telle manière à éviter les situations où la balle rebondit perpétuellement d'une raquette à l'autre sans avoir besoin de déplacer ces dernières.

### 3 Scores

Cacher le lutin

Si Joueur 1 gagne alors :

Ajouter 1 à **Score\_Joueur\_1**

Sinon

Ajouter 1 à **Score\_Joueur\_2**

Fin du test

Si **Score\_Joueur\_1** = 7 ou **Score\_Joueur\_2** = 7

Envoyer le message *Fin de partie*

Sinon

Envoyer le message *Nouvelle balle*

Fin du test

Algorithme

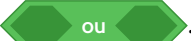
On quitte la boucle précédente lorsque la balle sort des limites du jeu. Il s'agit donc de mettre à jour le score du joueur qui a gagné et de vérifier si l'un des scores est égal à 7, ce qui signifie la fin de la partie.

Le Joueur 1 est celui dont la raquette est située à gauche de la scène tandis que le Joueur 2 est celui dont la raquette est à droite. Ainsi, si la balle sort à droite de la scène, c'est le score du Joueur 1 qu'il convient d'augmenter sinon, c'est celui du Joueur 2. Pour déterminer de quel côté la balle est sortie, on utilise le bloc **abscisse x**.

Aide

**abscisse x**

Ce bloc renvoie l'abscisse actuelle du lutin dans lequel il est employé.

Pour tester si le score de l'un des joueurs atteint 7 points, on utilise le bloc .

Aide



Ce bloc permet de tester si au moins l'une des deux conditions est remplie.

### 4 La vitesse de la balle

Un second script associé au lutin Ball permet d'augmenter progressivement la vitesse de la balle. Ce script est exécuté en même temps que le script qui gère le mouvement de la balle et se voit déclenché à la réception du message *Vitesse balle*.

Initialiser **Vballe** à 10

Répéter jusqu'à couleur noire touchée :

Attendre 5 secondes

Ajouter 1 à **Vballe**

Fin de la boucle

Algorithme

## 5 Améliorations possibles

Le programme, bien que jouable, peut encore être amélioré de multiples façons possibles. En particulier, on peut :

- afficher le nom de vainqueur;
- permettre de relancer une nouvelle partie à l'appui sur une touche;
- faire en sorte que la balle s'élance aléatoirement vers l'un ou l'autre des deux joueurs;
- créer un écran de présentation et un écran d'aide qui rappelle le principe du jeu et les touches actionnables;
- ajouter des sons;
- introduire d'autres lutins qui viendront modifier le cours de la partie (en augmentant ou en réduisant la taille des raquettes, en cachant la partie centrale de la scène, en ralentissant la vitesse de la raquette adverse, ...).

### À retenir

- Dans les programmes complexes, il est souvent intéressant de regrouper les différentes instructions qui permettent d'initialiser le programme dans le script associé à la scène.
- Pour donner l'illusion qu'un lutin se déplace à travers la scène, le principe consiste à modifier légèrement ses coordonnées en ajoutant ou en retranchant une certaine valeur (qui contribue à la « vitesse » du lutin) à son abscisse et/ou à son ordonnée.