**Atelier Saut**

**Objectifs:**

* Assurer le saut d’une entité (dans notre cas personnage)
* Si on clique sur la touche UP du clavier, le personnage saute, il change donc de position en fonction du temps.

1. **Saut vertical :**

Considérons la situation suivante : Un personnage se met au point A et doit sauter afin d’arriver au point B (voir figure ci-dessous).

**Description:**

* + - 1. Ajouter le champ up à la structure personnage **int up ;**

Ce champ sert à stocker l’état du personnage :

Si le personnage est en mouvement (saut) up vaut 1

Sinon up égale à 0.

* + - 1. Initialiser le champ up à 0 dans la fonction **void initPersonnage(Personnage \*P) ;**

**P->up=0 ;**

* + - 1. Implémenter la fonction du saut de la voiture **void saut(Personnage \*P,int dt, int posinit**) en fonction du temps dt et de la position posinit avec posinit le y initial du personnage
* **La montée du personnage**

-Si le champ up est activé (en état de montée) et la position du personnage n’a pas dépassé une certaine limite, comme déjà mentionné au niveau de la figure ci-dessous :

1. Saut vertical x



seuil [limite à ne pas dépasser ]





y 

**posinit**

**NB : seuil une valeur utile pour préciser la limite à ne pas dépasser.**

* **La descente du personnage**

Si la position du personnage atteint le seuil :

* Le champ up est en état de descente

Si le personnage est en train de descendre et son ordonné n’a pas encore atteint l’ordonné initial :

- On met à jour la position du personnage en fonction de la vitesse.

Si la position du personnage atteint la position initiale

* On désactive le champ up.

Au niveau de la boucle de jeu :

Programmer l’événement du saut.

Initialisation

…..

Début boucle du jeu

….

Lire évènement

Si l’évènement est un clic sur le bouton up alors

Faire appel à la fonction permettant au personnage de sauter

Fin de la boucle du jeu

Libérer personnage

1. **Saut parabolique :**

L’objectif est de faire sauter le personnage selon une fonction parabole.

Diagram

Description automatically generated

**Etape 1 :** Trouver l’équation de la parabole.

Une parabole a pour équation : y = ax²+c.

**Etape 2 :** Trouver les *coefficients aet c*de cette équation. En effet, si on connaît l'équation de notre parabole, on pourra donner le **déplacement horizontal** à notre programme pour qu'il calcule le **déplacement vertical**.

**2.2. Trouver c**

Le coefficient c est appelé ‘ ordonnée à l'origine’. Il représente *l'ordonnée du sommet de la parabole*. Cette ordonnée est définie par la hauteur du saut.

Dans l’exemple réalisé, on a fait sauter Mario de 100 pixels. **Donc** : c = 100.

**2.3. Trouver a**

Sachant b et c, l’équation de la parabole devient :

y = a\*x²+100

* Si a est positif, la parabole est tournée vers le haut, et inversement ;
* La parabole devient de plus en plus large quand a devient de plus en plus petit.

**NB:** Pour le premier point, a doit être **négatif**. En effet, on ne veut pas que notre personnage tombe et remonte, mais qu'il saute et qu'il redescende.

- Exemple de parabole tournée vers le haut, où a est par conséquent positif :

Diagram

Description automatically generated

- Et une tournée vers le bas, où a est négatif :

Chart, line chart

Description automatically generated

Soient les positions initiales de Mario sur le repère ‘réel’.

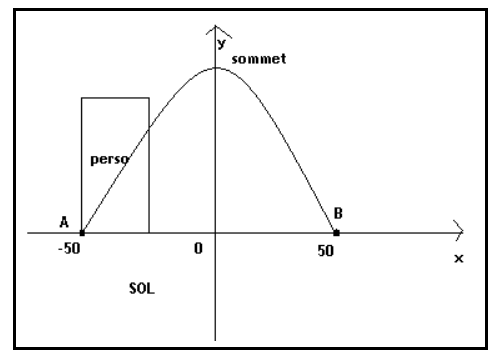
pos\_hero.x = 10;

pos\_hero.y = 250;

Pour réaliser le saut, on a besoin d’un deuxième repère (repère relatif ) dans lequel nous allons initialiser les positions initiales et finales du saut :

A(-50;0)  
B(50;0)

Cette figure représente les coordonnées dans le repère fictif :



Une fois on a initialisé les points de départ et d’arrivé, l’équation devient :

y = a\*x²+100.

On peut remplacer x et y par les coordonnées de l'un de ces points. Si on choisit A, on aura :

0 = a\*(-50)² + 100 ; on résout l'équation :  
2500a = -100.

**Donc** : a = -0,04.

**Etape 3 :** Implémentation de la fonction saut :

void saut\_Personnage ( Personnage \*P, Uint32 dt, int posx\_absolu, int posy\_absolu)

{

Si ( P->up == 1 ) alors  **// Si le personnage a fait un saut**

{

Mise à jour de la position du personnage selon l’axe relatif des abscisses

- l’absice du personnage change en fonction de la vitesse. A picture containing text

Description automatically generated

-l’ordonnée change en fonction de x avec la formule: y =ax²+ c

}

si y == posy

P->up = 0

}