

# PROJET 1: PROGRAMMATION DU JEU DE JUMPMAN

---

## 1. But

Le but de ce projet est de programmer le 1<sup>er</sup> tableau du jeu de **Jumpman**.

**Ce travail doit OBLIGATOIREMENT être réalisé en équipe de deux.**

## 2. Code Javascript

Étant donné que ce travail est en équipe, je vous suggère fortement de placer votre code **Javascript** dans plusieurs fichiers **.js** afin de bien répartir la tâche entre chaque étudiant (mais ce n'est pas obligatoire).

## 3. Remise du projet

- La date de remise est **Vendredi, le 18 mars 2016 à 23:59:59** (le lendemain de l'examen).
- Dans le titre de la page Web, vous devez écrire **Jumpman** suivi de vos noms.
- Remettre **votre jeu complet zippé** sur **LÉA** (avec les images s'il y en a ainsi que les sons et la musique).
- Ce projet comptera pour 13% de la session.
- Tout projet remis en retard sera pénalisé de 10% par jour de retard. Maximum 5 jours.
- **Si je m'aperçois que vous avez utilisé un jeu de **Jumpman** déjà programmé soit à partir d'Internet, soit à partir d'une autre équipe, vous obtiendrez automatiquement la note de 0%.**

## 4. Origine du jeu<sup>1</sup>

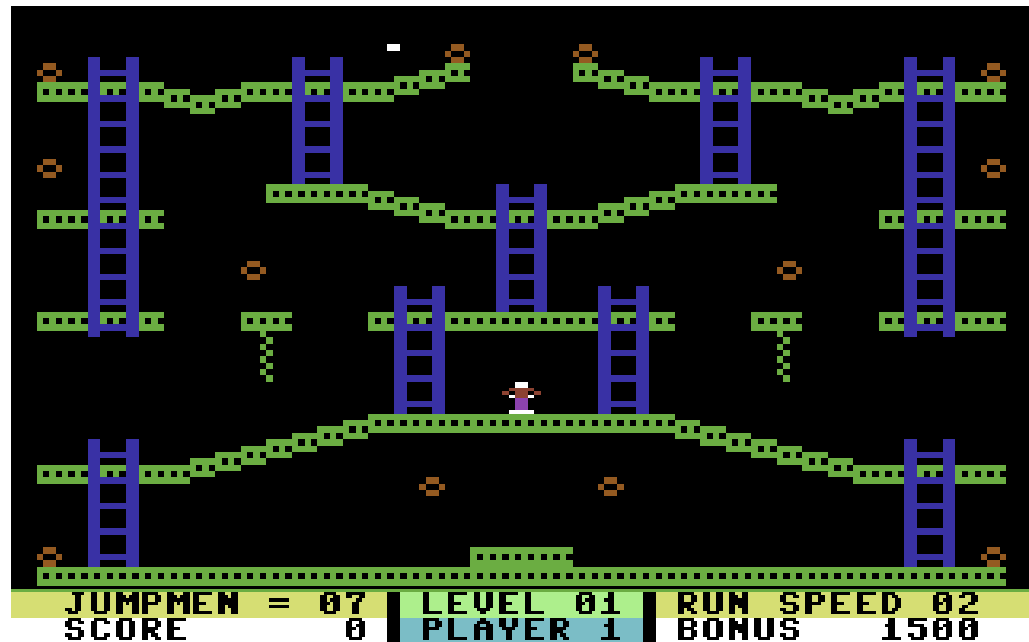
Au début de 1983, **Randy Glover** a développé le premier prototype du jeu de **Jumpman** pour l'**Atari 400**. Par la suite, il a recherché un éditeur pour commercialiser ce jeu et a signé avec **Epyx**. À la suite de cette signature, il a développé une version finale pour l'**Atari** et une version pour le **Commodore 64**. **Epyx** a également créé une version pour l'**Apple II**, et, un an plus tard, contracté **Mirror Images Software** pour une version pour **IBM PC jr**. Les versions **Atari** et **Commodore** ont été installées sur des disques et sur des cassettes tandis que les versions **Apple** et **IBM** ont été installées seulement sur des disques. Dans le passé, **Jumpman** a été considéré comme étant un jeu addictif et est devenu un best-seller pour **Epyx**, qui a vendu toutes les versions du jeu jusqu'en 1987.

## 5. Déroulement du jeu (le 1<sup>er</sup> tableau)<sup>2</sup>

Le but est de désamorcer toutes les bombes situées à des endroits déterminés sur des plates-formes et dans les airs. **Jumpman** désamorce une bombe en la touchant.

**Jumpman** peut se déplacer horizontalement sur une plate-forme, monter et descendre des échelles, grimper à l'aide d'une corde et bien sûr sauter. Les dangers du 1<sup>er</sup> tableau sont « tomber en bas d'une plate-forme » et « se faire frapper par une balle intelligente ».

Lorsque **Jumpman** meurt, il dégringole jusqu'en bas tout en rebondissant sur chacune des plates-formes situées en dessous.



<sup>1</sup> Référence : <https://en.wikipedia.org/wiki/Jumpman>

<sup>2</sup> Référence : <https://en.wikipedia.org/wiki/Jumpman>

## 6. Les objets de base du 1<sup>er</sup> tableau

Voici les objets dessinables que vous aurez à utiliser pour programmer votre jeu de **Jumpman** : les plates-formes, les échelles, les cordes, les balles intelligentes, les bombes et **Jumpman**.

### 6.1 Les plates-formes

Les plates-formes sont des structures fabriquées à partir de rectangles verts (ou d'une autre couleur si vous le voulez). Chacune des rectangles est percée de 2 petits trous.



Dans le 1<sup>er</sup> tableau, il y a 5 étages de plates-formes et, sur certaines plates-formes, il y a des escaliers et des trous.

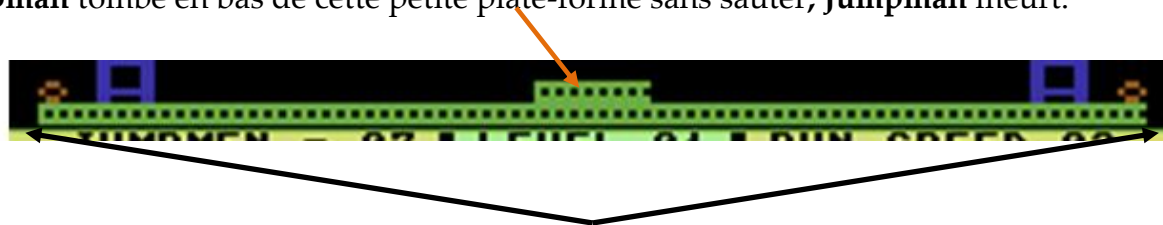
Pour déplacer **Jumpman** sur une plate-forme, l'utilisateur doit :

- 1- Appuyer sur la flèche-à-droite : Déplacer **Jumpman** d'un pas vers la droite
- 2- Appuyer sur la flèche-à-gauche : Déplacer **Jumpman** d'un pas vers la gauche
- 3- Appuyer sur la flèche-en-haut : Faire sauter **Jumpman**

La direction du saut dépend de la direction de **Jumpman** : si **Jumpman** est stationnaire, il saute en haut, si **Jumpman** se dirige vers la droite, il saute à droite et si **Jumpman** se dirige vers la gauche, il saute à gauche.

La hauteur et la longueur du saut dépendent du nombre de pas que **Jumpman** a faits auparavant. Par exemple, si **Jumpman** est stationnaire, fait trois pas vers la droite puis qu'il saute, il va sauter plus haut et plus loin que s'il n'avait fait qu'un seul pas.

**Note 1** : Si **Jumpman** tombe en bas de cette petite plate-forme sans sauter, **Jumpman** meurt.



**Note 2** : Si **Jumpman** tombe en bas de cette longue plate-forme, **Jumpman** meurt.

## 6.2 Les échelles

Pour se déplacer d'un étage à l'autre, **Jumpman** doit utiliser une échelle. Dans le 1<sup>er</sup> tableau, il y a 9 échelles : 2 grandes et 7 petites. Ces échelles sont situées à des endroits précis.

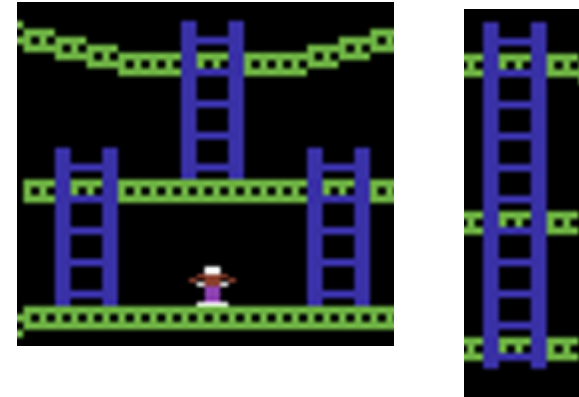
Pour atteindre l'échelle, **Jumpman** peut se déplacer jusqu'à l'échelle ou sauter sur l'échelle (dans ce cas, il reste accroché à l'échelle).

Pour déplacer **Jumpman** situé sur une échelle, l'utilisateur doit :

- 1- Appuyer sur la flèche-à-droite : Faire sauter **Jumpman** à la droite
- 2- Appuyer sur la flèche-à-gauche : Faire sauter **Jumpman** à la gauche
- 3- Appuyer sur la flèche-en-haut : Déplacer **Jumpman** vers le haut de l'échelle
- 4- Appuyer sur la flèche-en-bas : Déplacer **Jumpman** vers le bas de l'échelle

Lorsque **Jumpman** est rendu en haut de l'échelle, la touche flèche-en-haut fait sauter **Jumpman**.

Sur l'échelle, lorsque l'utilisateur fait sauter **Jumpman** vers la droite ou vers la gauche, **Jumpman** ne meurt pas s'il y a une plate-forme située juste en dessous. Il ne fait qu'atterrir sur la plate-forme. Par contre, s'il n'y a pas de plate-forme située juste en dessous (s'il y a un trou), **Jumpman** meurt.



## 6.3 Les cordes

Pour monter d'un étage à l'autre, **Jumpman** peut également grimper sur une corde. Dans le 1<sup>er</sup> tableau, il y a 2 cordes situées à des endroits précis.

Pour atteindre la corde, **Jumpman** doit être stationnaire juste en dessous puis sauter (vers le haut). Dès que **Jumpman** atteint la corde, il grimpe sur la corde de manière automatique jusqu'à ce qu'il atteigne la plate-forme. Pendant que **Jumpman** grimpe sur la corde, **Jumpman** peut sauter plus haut sur la corde (pour grimper plus vite) mais il ne peut pas aller vers la droite ni vers la gauche.

**Jumpman** ne peut pas utiliser la corde pour descendre. La corde ne sert qu'à monter.



## 6.4 Les balles intelligentes

Visuellement, une balle intelligente est un très petit rectangle blanc.

Parfois, au hasard, une balle se déplace en ligne droite sur l'écran. Au point de départ, elle provient soit de la gauche, soit de la droite, soit du haut, soit du bas de l'écran. Elle se déplace très lentement.



Dès que la balle est située sur la même ligne horizontale ou sur la même ligne verticale que **Jumpman**, cette première change brusquement de vitesse (elle va beaucoup plus rapidement) et fonce en ligne droite vers la position actuelle de **Jumpman**. Il est possible que la balle intelligente ne détecte pas **Jumpman**.

De deux choses l'une :

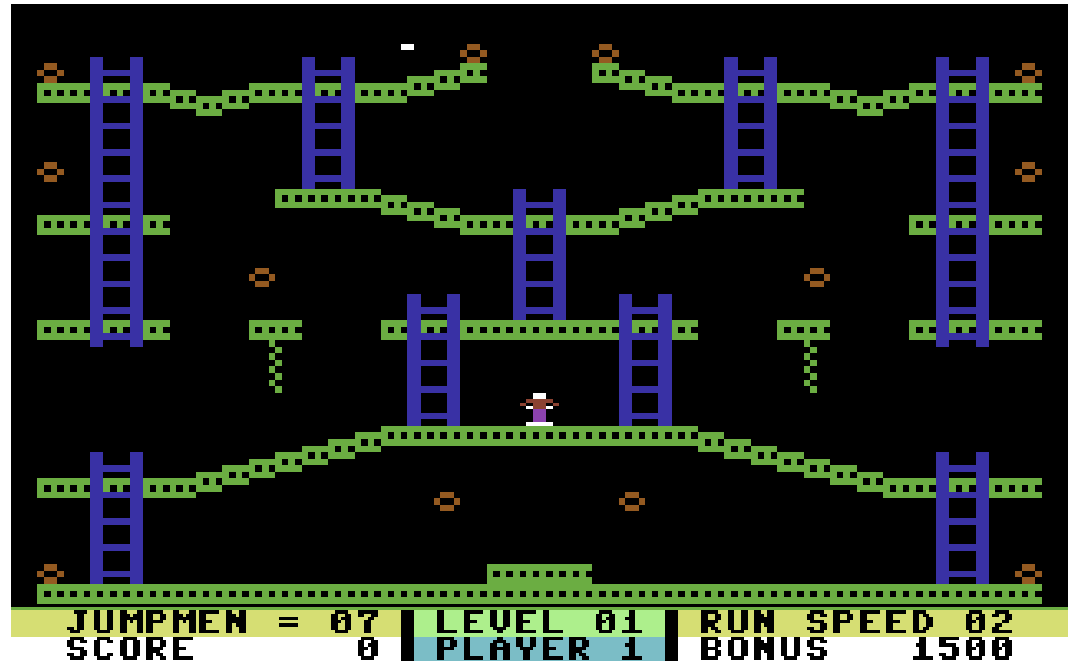
- la balle atteint **Jumpman** ; dans ce cas, **Jumpman** tombe et meurt,
- la balle n'atteint pas **Jumpman** (car **Jumpman** s'est déplacé entre-temps) : dans ce cas, elle continue sa course jusqu'à ce qu'elle disparaisse de l'écran.

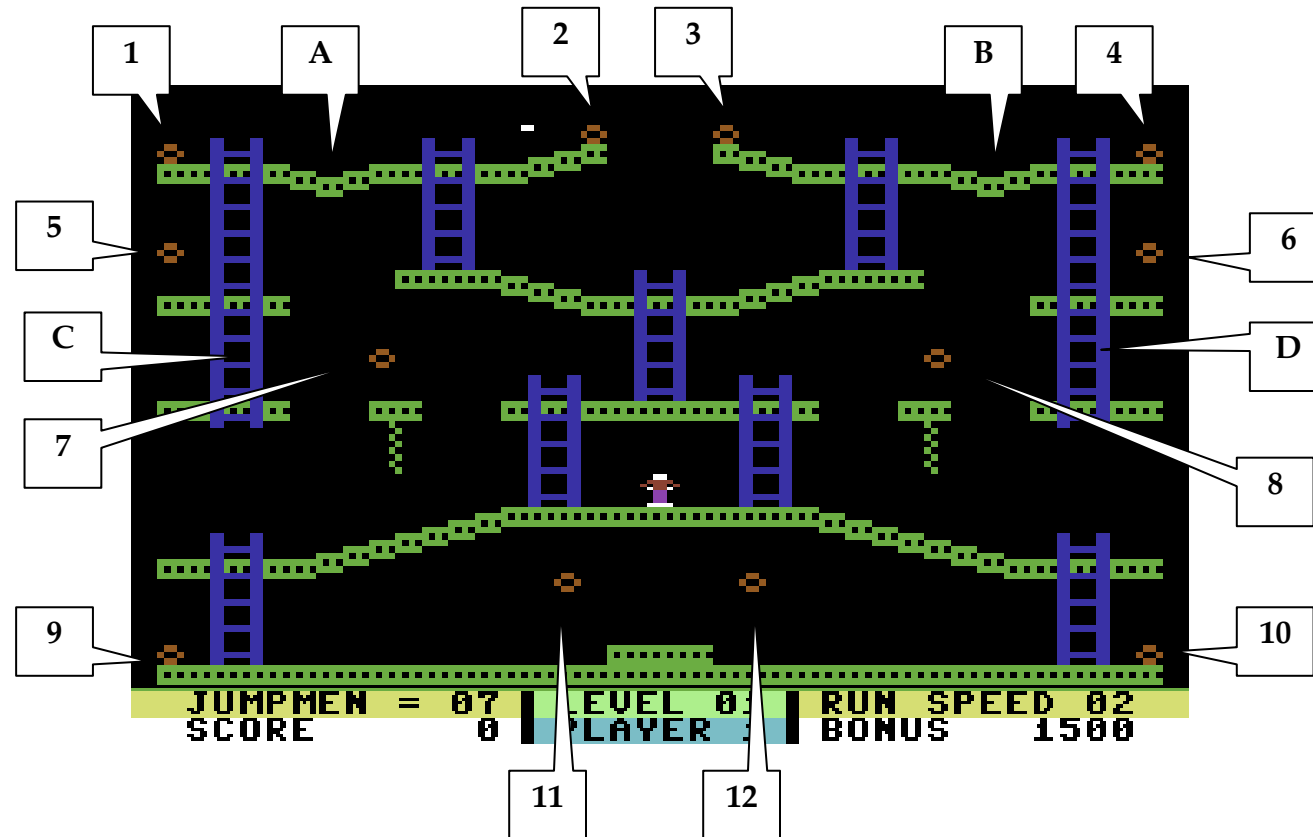
## 6.5 Les bombes

Dans le 1<sup>er</sup> tableau du jeu, il y a 12 bombes situées à des endroits précis.

Lorsque **Jumpman** entre en collision avec une bombe, celle-ci se désactive et disparaît.

Lorsqu'elles sont désactivées, certaines bombes ont un effet sur les plateformes ou sur les échelles.

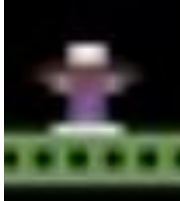



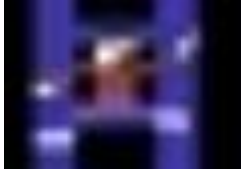


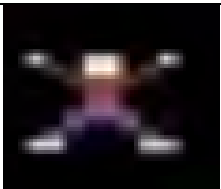









- Bombe 1 désactivé : Les 3 rectangles verts situés en **A** disparaissent (formant un trou sur la plate-forme)
- Bombe 4 désactivé : Les 3 rectangles verts situés en **B** disparaissent (formant un trou sur la plate-forme)
- Bombe 5 désactivé : Une partie de la grande échelle située en **C** disparaît (formant un trou dans l'échelle; cela empêche **Jumpman** d'utiliser cette partie de l'échelle pour monter ou descendre; s'il le fait, il tombe et meurt)
- Bombe 6 désactivé: Une partie de la grande échelle située en **D** disparaît (formant un trou dans l'échelle; cela empêche **Jumpman** d'utiliser cette partie de l'échelle pour monter ou descendre; s'il le fait, il tombe et meurt)
- Bombes 7 et 8 : **Jumpman** doit utiliser la corde ou sauter à partir de l'échelle pour atteindre la bombe
- Bombes 11 et 12 : **Jumpman** doit sauter pour atteindre la bombe

## 6.6 Les différents aspects de **Jumpman**

**Jumpman** prend différents aspects selon son action (vous devez dessiner **Jumpman**)

<b>Jumpman</b> est stationnaire		<b>Jumpman</b> se déplace vers la gauche	 
<b>Jumpman</b> saute vers la droite		<b>Jumpman</b> descend ou monte une échelle	
<b>Jumpman</b> saute vers la gauche			
<b>Jumpman</b> saute vers le haut			
<b>Jupman</b> grimpe sur la corde	   		

## 7. Les sons du jeu

Au début du tableau ou après avoir perdu une vie, une courte musique se fait entendre; pendant ce temps **Jumpman** apparaît progressivement au centre du 2<sup>ième</sup> étage. Aucune touche du clavier ne fonctionne pendant tout ce temps.

Lorsque **Jumpman** se déplace sur les plates-formes ou sur l'échelle, on entend ses pas.

Lorsque **Jumpman** saute, on entend un son indiquant qu'il est en train de sauter. Aucune touche du clavier ne fonctionne pendant tout ce temps.

Lorsque **Jumpman** est détecté par une balle intelligente, on entend un coup de fusil.

Lorsque **Jumpman** désactive une bombe, on entend un son indiquant que la bombe est désactivée.

Lorsque **Jumpman** tombe en bas d'une plate-forme, on l'entend tomber et rebondir et, à la fin de sa chute, on entend une musique de défaite. Aucune touche du clavier ne fonctionne pendant tout ce temps.

Lorsque **Jumpman** désactive la 12<sup>ième</sup> bombe, le jeu se fige complètement et on entend une musique de victoire. Aucune touche du clavier ne fonctionne pendant tout ce temps.

## 8. Le nombre de vies, le score et le bonus

Dans le jeu original, **Jumpman** a 7 vies. Dans votre jeu, limitez le nombre de vies à 3.

Dans le jeu original, l'utilisateur peut choisir le nombre de joueurs. Ne le faites pas.



JUMP MEN = 07	LEVEL 01	RUN SPEED 02
SCORE 0	PLAYER 1	BONUS 1500

À chaque fois que **Jumpman** désactive une bombe, le score du joueur doit être augmenté de 100 points.

De plus, le joueur peut obtenir un bonus en fonction du temps qu'il a pris pour terminer le tableau. Au point de départ, le bonus est de 1500 points et diminue de 100 points à tous les 5 secondes. Par exemple, si le joueur prend 15 secondes pour terminer le tableau, il obtient 1200 points de bonus. À la fin du tableau, ces points bonus doivent automatiquement s'ajouter à son score.



## 9. La vitesse de **Jumpman**

Dans le jeu original, l'utilisateur peut choisir la vitesse de **Jumpman**. Ne le faites pas. Au point de départ, la vitesse de **Jumpman** doit être de 1.

Dans le jeu original, lorsque **Jumpman** a perdu toutes ses vies, le joueur doit recommencer le jeu au complet et, s'il a réussi un tableau, il doit faire le tableau suivant.

JUMP MEN = 07	LEVEL 01	RUN SPEED 02
SCORE 0	PLAYER 1	BONUS 1500

Étant donné que vous allez programmer seulement le 1<sup>er</sup> tableau, lorsque **Jumpman** a perdu ses 3 vies, recommencez le jeu au complet (remettez le score à 0, le nombre de vies à 3 et la vitesse à 1).

Par contre, si **Jumpman** n'a pas perdu ses 3 vies, recommencez le même tableau sans modifier le score ni le nombre de vies mais augmentez la vitesse de **Jumpman**.

Pour augmenter la vitesse de **Jumpman**, vous n'avez qu'à augmenter le nombre de pixels que vous devez ajouter à la position de **Jumpman**. Par exemple, à la vitesse 1, **Jumpman** pourrait se déplacer de 4 pixels à la fois; à la vitesse 2, **Jumpman** pourrait se déplacer de 6 pixels à la fois, à la vitesse 3, **Jumpman** pourrait se déplacer de 8 pixels à la fois, etc...

Faites des tests concernant la vitesse de **Jumpman**.

## 10. Contraintes de programmation

- Tous vos objets doivent être **dessinés incluant les différents aspects de Jumpman**. Je ne veux voir aucune image (sauf, possiblement, une image de fond). Il n'est nécessaire que vos dessins soient identiques à l'original mais plus beaux seront vos dessins, mieux vous serez notés.
- Vous devez programmer votre jeu à l'aide de cycles d'animation (comme cela a été fait dans le jeu du vortex).
- Vous devez programmer chacun des éléments du jeu dans des objets distincts.
- Vous devez avoir, au minimum, 8 type d'objets :
  - 1 objet de type **plates-formes** (la taille des rectangles verts et un tableau qui indique la position de chacun d'eux).
  - 9 objets de type **échelle**,
  - 2 objets de type **corde**,
  - 1 objet de type **balle**,
  - 12 objets de type **bombe**,
  - 1 objet de type **Jumpman**
  - un objet de type **sons** (qui contient tous les sons et toutes les musiques du jeu)
  - un objet qui décrit l'état du jeu à un certain moment (dans lequel vous pouvez mettre le score, le nombre de vies restantes, le bonus actuel, si les touches du clavier sont désactivées ou non et toute autre information nécessaire au bon déroulement du jeu).
- Chacun des vos objets doit être complet. Les propriétés doivent décrire son état à n'importe quel moment du jeu.

Par exemple, dans l'objet de type **Jumpman**, vous pouvez mettre sa position (x,y) sur l'écran, sa taille, sa direction, sa vitesse, le nombre de pas qu'il a fait dans une direction donnée, le nombre de pixels à laquelle il se déplace, la longueur du saut, la hauteur du saut, un indicateur qui indique l'aspect de **Jumpman** (cela va être utilisé pour dessiner **Jumpman**), etc.

Par exemple, dans un objet de type **bombe**, vous pouvez mettre sa position (x,y) sur l'écran, sa taille, un booléen qui indique si la bombe est désactivée ou non, etc.

- Prévoyez des fonctions pour créer et initialiser chacun des objets.
- Prévoyez des fonctions pour dessiner chacun des objets dessinables. CETTE FONCTION NE DOIT JAMAIS MODIFIER L'ÉTAT DE CET OBJET NI DES AUTRES OBJETS. ELLE DOIT DESSINER L'OBJET. C'EST TOUT.
- L'état des objets doit être modifié à deux endroits spécifiques : dans l'événement du clavier et dans la fonction qui met à jour l'animation.
- À part des circonstances exceptionnelles, n'appellez JAMAIS directement la fonction qui dessine un objet ni celle qui met à jour de l'animation. C'est le cycle d'animation qui doit s'en occuper.

## 11. Démonos

Je n'ai pas trouvé de jeu **Jumpman** sur Internet. Par contre, j'ai trouvé des vidéos sur **Youtube** de **Jumpman** sur le **Commodore 64**.

Voici les liens :

<https://www.youtube.com/watch?v=FXjOOaQPpcg> (dans cette vidéo, on voit ce qui se passe lorsque **Jumpman** meurt)

<https://www.youtube.com/watch?v=HViFuRD5I64>

<https://www.youtube.com/watch?v=dCdLnA7f3Hc>

<https://www.youtube.com/watch?v=dVQLfbyp-xs>

<https://www.youtube.com/watch?v=rc1stgFZrQ4>

<https://www.youtube.com/watch?v=RFCpvg9qxE>

[https://www.youtube.com/watch?v=KEal\\_HfcM5c](https://www.youtube.com/watch?v=KEal_HfcM5c)

<https://www.youtube.com/watch?v=3UXb8Bm19Io>

## 12. Correction

Voici la pondération pour la correction :

- Le bon déroulement du jeu ainsi que le respect des règles du jeu : 80%
- L'interface : Bonne vitesse, fluidité des animations, qualité des dessins, ressemblance avec l'original: 20%
- Non respect des contraintes de programmation : jusqu'à -20%