

Projet Arduino : RUNNER ISEN

Par groupe de 4 avec 2 cartes par groupe sur 4h (deux séances de 2h).

Objectif :

Faire une application interactive avec la carte arduino ISEN de type runner. Cette activité permet de contrôler un personnage avec un unique bouton de saut. Le personnage représente un coureur et reste à la même place sur l'affichage tandis que le paysage défile à l'horizontal. Il faut esquiver les obstacles à l'aide du saut afin de parcourir la plus grande distance possible.

Exemple de runner un peu plus complexe : <https://chromedino.com/>



Matériel :

Le matériel nécessaire pour ce projet est intégralement présent sur les cartes ISEN mises à disposition (un bouton poussoir, une led, un écran lcd et un arduino).

Sous-objectifs :

1. Le personnage (toujours dans la deuxième colonne) :
 - a. Implémenter de nouveaux caractères affichables sur le lcd pour représenter un personnage changeant de ligne en plusieurs étapes et des obstacles qui se déplacent horizontalement. Il faut prévoir ici une séquence de montée et une séquence de descente distinctes.
 - b. Prévoir le contrôle du personnage par un bouton utilisé en mode interruption. La séquence de saut ne doit se déclencher que si le personnage est situé au sol sur la ligne du bas lors du déclenchement de l'interruption.
2. Le décor :
 - a. Il s'agit d'une séquence d'obstacles générée aléatoirement mais telle qu'il soit possible de les éviter en faisant des sauts.
 - b. Le décor doit défiler à l'écran de droite à gauche.
 - c. Les animations (saut et défilement) sont basées sur une boucle mettant à jour l'affichage. Il faut faire accélérer le défilement régulièrement depuis une vitesse initiale jusqu'à une vitesse maximale.
3. L'interface de l'application :
 - a. Créer une application ayant deux modes, les mode « attente » et « en jeu » et permettre de passer de l'un à l'autre.
 - b. Le mode attente peut être fixe ou animé et présente le jeu sur l'écran lcd (personnage, obstacle, titre ou mots clés comme RUNNER/ISEN/JUNIA/PRESS START ...) jusqu'à ce qu'un appui sur le bouton déclenche le changement de mode.
 - c. Le mode jeu s'exécute jusqu'à ce que vous détectiez une collision entre le personnage et un obstacle, le jeu est alors perdu et reviens en mode attente.
 - d. Prévoyez un indicateur pour le joueur en allumant une led quand c'est le bon moment pour sauter.
 - e. Détecter la fin d'un obstacle ou son absence pour déclencher la séquence de descente du personnage.

4. Créativité et originalité (objectifs bonus) par exemple:

- a. Comptabiliser un score avec la distance parcourue et conserver le dernier pendant le mode attente.
- b. Prévoir d'autres obstacles ou interactions.
- c. Prévoir plusieurs modes de difficulté (vitesse initiale/max/accélération, distance et taille des obstacles, durée de suspension du personnage en haut de son saut ...).
- d. Qualité du mode attente.

Conseils :

- Le jeu sera basé sur une boucle contenant une instruction delay(XX). Vous devez associer une seule image (16*2 caractères) par boucle et la succession des boucles donnera l'impression de mouvement. En diminuant la valeur de XX vous pouvez faire accélérer le jeu.
- Prévoyez une séquence d'images assez courte pour la montée du personnage et une à plusieurs étapes de suspension avant de commencer à vérifier s'il doit redescendre. Il faut de plus qu'entre deux obstacles il y ait le temps de faire une séquence de descente plus une séquence de montée complète.
- L'écran lcd peut mémoriser jusqu'à 8 caractères modifiables (de 16*5 pixels) numérotés de 0 à 7. Vous pouvez créer un caractère pixel par pixel de la manière suivante :

Pour définir le caractère :

```
byte smiley[8] ={  
  B00000,  
  B00000,  
  B01010,  
  B00000,  
  B00000,  
  B10001,  
  B01110,  
  B00000  
};
```

*Pour associer le caractère à l'écran
en tant que caractère libre numéro
2 :*

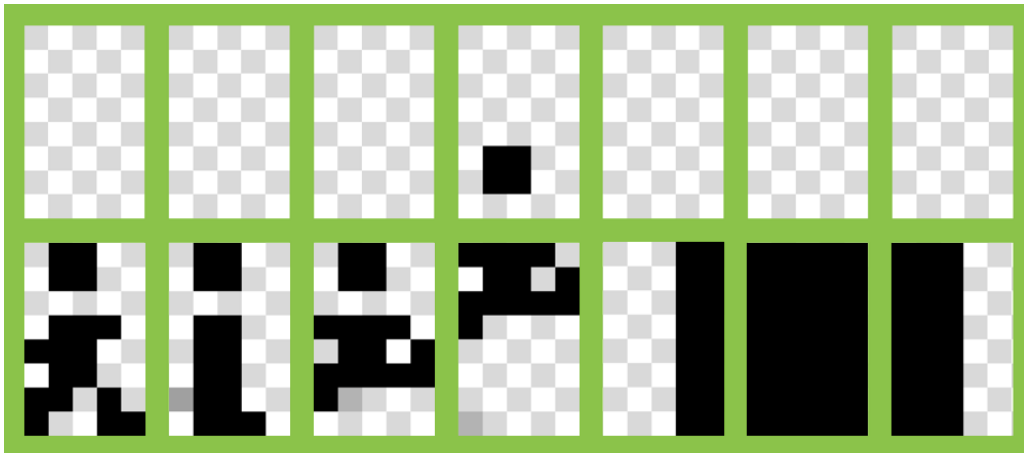
```
lcd.createChar(2,smiley);
```

*Pour afficher le caractère à la
position actuelle du curseur :*

```
lcd.write(2);
```

Le caractère 0 peut poser problème, il faut alors écrire lcd.write(byte(0)) ;

Vous pouvez utiliser les caractères suivants ou créer les vôtres :



Vous pouvez représenter la course en alternant les images 1 et 2, une séquence de montée avec les images 1, 3, 4 (sur deux lignes donc deux images) puis 3 (en suspension en haut) et enfin 1 (cours sur la ligne du haut) et une descente avec 3(haut) , 4, 3(bas), 1. Ci-dessous, une représentation de la descente en ne comportant que la colonne 2 du lcd sur 5 pas de temps successifs.



- Pour faire de l'aléatoire avec arduino, initialisez le générateur de nombre aléatoire dans le setup avec l'instruction : `randomSeed(analogRead(0))` puis utiliser la fonction `random(val)` dans la suite de votre code. `Random (val)` donne une valeur entre 0 et val mais il est possible de donner deux arguments en guise de minimum(inclus) et maximum(exclu).
- Pour le défilement du décor, vous pouvez faire une boucle sur chaque caractère du lcd et en fonction de ce qu'elle contient et de ce que contient la case immédiatement à sa droite vous pouvez déterminer sa prochaine valeur. Pour la dernière case vous pouvez par exemple mémoriser le nombre d'étapes écoulées depuis le passage du dernier obstacle et lorsque ce nombre est suffisant tirer aléatoirement l'apparition d'un nouveau block en fonction d'une probabilité que vous aurez définie.