

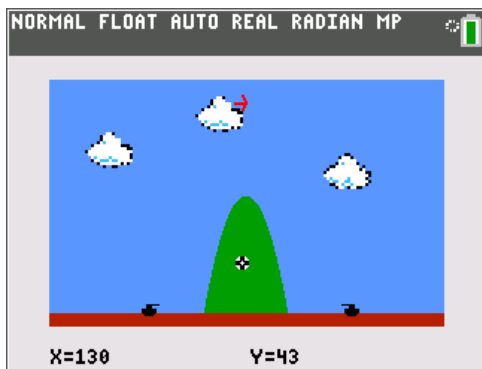
# LE MOTEUR DE JEU

TI-83 Premium CE

## 1. Objectifs

- Construire le moteur de notre jeu (interaction et tir), en utilisant les primitives graphiques spécifiques à la TI-83 Premium CE dans l'environnement de programmation de la calculatrice.
- La trajectoire de tir sera parabolique et sera impactée par la direction et la force du vent.
- Ce projet sera l'occasion de comprendre l'usage des variables globales dans l'environnement de programmation.
- Il s'agit de la troisième activité d'un ensemble de trois activités aboutissant à la programmation d'un jeu fonctionnel.

## 2. Énoncé



Maintenant que le décor est construit et les variables prêtes, nous allons, dans cette dernière activité, mettre en place les éléments d'interaction permettant de configurer le tir, le représenter et déterminer s'il s'agit ou non d'un tir gagnant.

## 3. Conduite de l'activité



Le moteur de jeu consistera en un seul programme appelant l'ensemble des programmes créés.

Le programme principal initialisera, après l'affichage du splashscreen, le dessin de la montagne, des tanks et du vent (reprise du travail précédent).

Il faudra :

- saisir l'angle et de la force du tir par le joueur qui a la main ;
- tracer la trajectoire du tir. Placer correctement le départ du tir en fonction de la position du canon.
- ensuite, selon si le tir est raté ou gagnant, effacer la trajectoire ou afficher un écran de victoire !

## 4. Commentaires

Pour démarrer la programmation de cette activité, il faut préalablement résoudre la question de la trajectoire du tir.

La trajectoire doit tenir compte du vent. Nous allons simplifier au maximum les conditions du tir en choisissant classiquement une valeur de 10 pour la gravité.

Nous utiliserons les équations suivantes, bien connues du programme de physique lorsqu'on étudie le déplacement parabolique d'un objet de coordonnées  $(x ; y)$  avec une vitesse initiale  $V_0$  d'angle  $\alpha$ , le tout exprimé en fonction du temps :

$$x(t) = V_0 \cos(\alpha) t$$

$$y(t) = V_0 \sin(\alpha) t - g/2 t^2$$

En tenant compte de la vitesse horizontale  $E$  du vent, nous avons donc :

$$x(t) = (E + V_0 \cos(\alpha)) t$$

$$y(t) = V_0 \sin(\alpha) t - g/2 t^2$$

Finalement, compte tenu des variables globales imposées et de la gravité égale à 10 :

$$x(T) = (E + F \cos(A)) T$$

$$y(T) = F \sin(A) T - 5 T^2$$

Il ne reste donc plus qu'à positionner correctement le tir pour qu'il quitte bien le canon du char d'assaut.

## 5. ANNEXE

- En vue de la poursuite de l'ensemble du projet, respecter les affectations de variables :

A : Angle de tir sélectionné par un joueur

D : Départ (position de) du tir

E : vitesse du vent (valeur comprise entre -5 et 5) (la variable V est déjà attribuée)

F : Force (vitesse) initiale du tir

G : Gagné ? (1 si oui, 0 sinon)

H : Hauteur de la montagne (valeur comprise entre 60 et 160)

J : Joueur en action (1 si joueur de gauche, -1 si joueur de droite)

L : Largeur de la montagne (valeur, par exemple comprise, entre 0,125 et 0,5)

O : Objectif (position de l')

P : Place disponible

T : Temps pour le calcul de la trajectoire du tir

U : Position tank de gauche

V : Position tank de droite

Y1 : équation de la montagne

- Respecter de même les noms de programmes suivants :

DInit.8XP : le programme de préparation de l'écran graphique.

DMontagn.8XP : le programme de dessin de la montagne.

DSplash.8XP : le programme de dessin de l'écran d'introduction du jeu.

DTank.8XP : le programme de dessin des chars d'assaut.

DVent.8XP : le programme de dessin de la force du vent.

DBoom.8XP : le programme qui dessine l'explosion du char d'assaut.

DTraject.8XP : le programme qui dessine la trajectoire du tir.

Joueur1.8XP : le programme qui gère la saisie de l'angle et de la force du joueur1.

Joueur2.8XP : le programme qui gère la saisie de l'angle et de la force du joueur2.

Tank.8XP : le moteur de jeu gérant l'ensemble des appels et procédures.