

T.P. 11

Space Invaders (partie 14)

Étape 1

Nous allons maintenant ajouter la petite animation des tirs d'envahisseurs. C'est-à-dire alterner continuellement les bitmaps 1 et 2.

Soit le sous-programme suivant qui échange les bitmaps des tirs d'envahisseurs :

```
SwapInvaderShots    ; Sauvegarde les registres.
                   movem.l d7/a1,-(a7)

                   ; Nombre d'itérations = Nombre de tirs d'envahisseurs.
                   ; Nombre d'itérations - 1 (car DBRA) -> D7.W
                   move.w #INVADER_SHOT_MAX-1,d7

                   ; Adresse des tirs d'envahisseurs -> A1.L
                   lea    InvaderShots,a1

\loop               ; Échange les bitmaps 1 et 2 pour tous les tirs.
                   jsr    SwapBitmap
                   adda.l #SIZE_OF_SPRITE,a1
                   dbra   d7,\loop

                   ; Restaure les registres puis sortie.
                   movem.l (a7)+,d7/a1
                   rts
```

Ajoutez un appel à **SwapInvaderShots** à la fin de **MoveInvaderShots** (juste avant la restauration des registres) et relancez votre programme principal. On constate que l'animation est trop rapide (à la limite du perceptible). Il va donc falloir la ralentir.

Pour ce faire, modifiez le sous-programme **SwapInvaderShots** de façon à ce que l'échange de bitmaps se fasse une fois sur six. C'est-à-dire que six appels à **SwapInvaderShots** sont nécessaires pour échanger les bitmaps (cinq appels sur six ne feront rien).

Relancez le programme principal et vérifiez que l'animation soit perceptible (cela peut varier selon la puissance de votre ordinateur). Si l'animation est toujours trop rapide, diminuer la fréquence de répétition (par exemple 1 fois sur 10).

Étape 2

Pour terminer la gestion des tirs d’envahisseurs, réalisez le sous-programme **IsShipHit** qui détermine si le vaisseau a été touché par un tir d’envahisseur.

Sortie : **Z** renvoie *false* (0) si le vaisseau n’a pas été touché par un tir d’envahisseur.

Z renvoie *true* (1) si le vaisseau a été touché par un tir d’envahisseur.

Indications :

Pensez à utiliser le sous-programme **IsSpriteColliding** déjà réalisé dans une étape précédente.

Modifiez ensuite votre programme principal de la façon suivante :

```

Main          jsr    InitInvaders
              jsr    InitInvaderShots

\loop
              jsr    PrintShip
              jsr    PrintShipShot
              jsr    PrintInvaders
              jsr    PrintInvaderShots
              jsr    BufferToScreen

              jsr    DestroyInvaders

              jsr    MoveShip
              jsr    MoveInvaders
              jsr    MoveShipShot
              jsr    MoveInvaderShots

              jsr    NewShipShot
              jsr    NewInvaderShot

              jsr    SpeedInvaderUp

              jsr    IsShipHit
              bne    \loop

              illegal
  
```

Lancez-le et vérifiez qu’il s’arrête lorsqu’un tir d’envahisseur atteint le vaisseau.

Étape 3

En vous inspirant de **IsShipHit**, réalisez le sous-programme **IsShipColliding** qui détermine si le vaisseau entre en contact avec un envahisseur.

Sortie : **Z** renvoie *false* (0) si le vaisseau n'est pas en contact avec un envahisseur.

Z renvoie *true* (1) si le vaisseau est en contact avec un envahisseur.

Indications :

Ce sous-programme est quasiment identique à **IsShipHit**.

Modifiez ensuite votre programme principal de la façon suivante :

```

Main          jsr    InitInvaders
              jsr    InitInvaderShots

\loop
              jsr    PrintShip
              jsr    PrintShipShot
              jsr    PrintInvaders
              jsr    PrintInvaderShots
              jsr    BufferToScreen

              jsr    DestroyInvaders

              jsr    MoveShip
              jsr    MoveInvaders
              jsr    MoveShipShot
              jsr    MoveInvaderShots

              jsr    NewShipShot
              jsr    NewInvaderShot

              jsr    SpeedInvaderUp

              jsr    IsShipColliding
              bne    \loop

              illegal

```

Lancez-le et vérifiez qu'il s'arrête lorsque le vaisseau entre en contact avec un envahisseur.

Étape 4

Réalisez le sous-programme **IsInvaderTooLow** qui détermine si un envahisseur descend trop bas sur l'écran. Nous considérerons qu'un envahisseur est trop bas lorsque son abscisse est supérieure à 280.

Sortie : **Z** renvoie *false* (0) si aucun envahisseur n'est trop bas.

Z renvoie *true* (1) si au moins un envahisseur est trop bas.

Modifiez ensuite votre programme principal de la façon suivante :

```

Main          jsr    InitInvaders
              jsr    InitInvaderShots

\loop        jsr    PrintShip
              jsr    PrintShipShot
              jsr    PrintInvaders
              jsr    PrintInvaderShots
              jsr    BufferToScreen

              jsr    DestroyInvaders

              jsr    MoveShip
              jsr    MoveInvaders
              jsr    MoveShipShot
              jsr    MoveInvaderShots

              jsr    NewShipShot
              jsr    NewInvaderShot

              jsr    SpeedInvaderUp

              jsr    IsInvaderTooLow
              bne    \loop

              illegal
  
```

Lancez-le et vérifiez qu'il s'arrête lorsqu'un envahisseur est trop bas.