

# T.P. 6 – Corrigé

## Space Invaders (partie 9)

### Étape 1

```

InitInvaderLine    ; Sauvegarde les registres.
                   movem.l d1-d3/d7/a0,-(a7)

                   ; Nombre d'itérations = Nombre d'envahisseurs par ligne
                   ; Nombre d'itérations - 1 (car DBRA) -> D7.W
                   move.w #INVADER_PER_LINE-1,d7

                   ; Modifie l'abscisse de départ de la ligne
                   ; afin de centrer le sprite sur une largeur de 32 pixels.
                   ; D1.W += (32 - Largeur du sprite) / 2
                   move.w #32,d3
                   sub.w  WIDTH(a1),d3
                   lsr.w  #1,d3
                   add.w  d3,d1

\loop              ; Initialise tous les champs du sprite.
                   move.w #SHOW,STATE(a0)
                   move.w d1,X(a0)
                   move.w d2,Y(a0)
                   move.l a1,BITMAP1(a0)
                   move.l a2,BITMAP2(a0)

                   ; Passe à l'envahisseur suivant.
                   adda.l #SIZE_OF_SPRITE,a0
                   addi.w #32,d1
                   dbra  d7,\loop

                   ; Restaure les registres puis sortie.
                   movem.l (a7)+,d1-d3/d7/a0
                   rts

```

```

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

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

                   ; Adresse de départ des sprites -> A1.L
                   lea  Invaders,a1

\loop              ; Affiche un envahisseur.
                   jsr  PrintSprite

                   ; Passe au prochain envahisseur et reboucle.
                   adda.l #SIZE_OF_SPRITE,a1
                   dbra  d7,\loop

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

```

## Étape 2

```
InitInvaders      ; Sauvegarde les registres.
                  movem.l d1/d2/a0-a2,-(a7)

                  ; 1re ligne d'envahisseurs.
                  move.w InvaderX,d1
                  move.w InvaderY,d2
                  lea Invaders,a0
                  lea InvaderC_Bitmap,a1
                  lea 0,a2
                  jsr InitInvaderLine

                  ; 2e ligne d'envahisseurs.
                  add.w #32,d2
                  adda.l #SIZE_OF_SPRITE*INVADER_PER_LINE,a0
                  lea InvaderB_Bitmap,a1
                  jsr InitInvaderLine

                  ; 3e ligne d'envahisseurs.
                  add.w #32,d2
                  adda.l #SIZE_OF_SPRITE*INVADER_PER_LINE,a0
                  jsr InitInvaderLine

                  ; 4e ligne d'envahisseurs.
                  add.w #32,d2
                  adda.l #SIZE_OF_SPRITE*INVADER_PER_LINE,a0
                  lea InvaderA_Bitmap,a1
                  jsr InitInvaderLine

                  ; 5e ligne d'envahisseurs.
                  add.w #32,d2
                  adda.l #SIZE_OF_SPRITE*INVADER_PER_LINE,a0
                  jsr InitInvaderLine

                  ; Restaure les registres puis sortie.
                  movem.l (a7)+,d1/d2/a0-a2
                  rts
```