

T.P. 8 – Corrigé

Space Invaders (partie 1)

Étape 1

```
FillScreen      ; Sauvegarde les registres dans la pile.
                movem.l d7/a0,-(a7)

                ; Fait pointer A0 sur la mémoire vidéo.
                ; Cette instruction est identique à : movea.l #VIDEO_START,a0
                lea    VIDEO_START,a0

                ; Initialisation du compteur de boucle (D7.W).
                ; La copie se fera sur 32 bits, c'est-à-dire sur 4 octets.
                ; Le nombre d'itérations est donc la taille en octets divisée par 4.
                ; Le test de sortie se fera à l'aide de DBRA,
                ; donc D7.W doit contenir le nombre d'itérations moins 1 (cf. cours).
                move.w #VIDEO_SIZE/4-1,d7

\loop           ; Copie la donnée dans la mémoire vidéo
                ; et passe à l'adresse suivante.
                move.l d0,(a0)+
                dbra   d7,\loop

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

Étape 2

```

HLines      ; Sauvegarde les registres dans la pile.
            movem.l d6/d7/a0,-(a7)

            ; Fait pointer A0 sur la mémoire vidéo.
            lea    VIDEO_START,a0

            ; D7.W = Compteur de boucle
            ;      = Nombre d'itérations - 1 (car DBRA).
            ; -----
            ; Nombre d'itérations = Nombre de rayures blanches et noires
            ; Hauteur d'un rayure blanche = 8 pixels
            ; Hauteur d'un rayure noire = 8 pixels
            ; Nombre de rayures blanches et noires = Hauteur de la fenêtre / 2x8
            move.l #VIDEO_HEIGHT/16-1,d7

\loop      ; Dessine une rayure blanche (8 lignes blanches).
            ; -----
            ; D6.W = Compteur de boucles
            ;      = Nombre d'itérations - 1 (car DBRA)
            ; -----
            ; Nombre d'itérations = Nombre de mots longs
            ; Nombre de mots longs = Nombre d'octets / 4
            ; Nombre d'octets      = BYTE_PER_LINE x Hauteur d'une rayure blanche
            ; Hauteur d'une ligne = 8 pixels
            move.w #BYTE_PER_LINE*8/4-1,d6
\white_loop move.l #$ffffffff,(a0)+
            dbra  d6,\white_loop

            ; Dessine une rayure noire (8 lignes noires).
            move.w #BYTE_PER_LINE*8/4-1,d6
\black_loop clr.l  (a0)+
            dbra  d6,\black_loop

            ; Reboucle tant qu'il reste des rayures
            ; blanches et noires à dessiner.
            dbra  d7,\loop

            ; Restaure les registres puis sortie.
            movem.l (a7)+,d6/d7/a0
            rts

```