

Rapport de projet

Mars Invaders - Architecture des Ordinateurs

Gaëlle GIRARD - Lucas LETT

Tout d'abord, nous avons tout deux dressé l'inventaire des différentes fonctions et des différentes données fournies dans le programme initial. Nous avons, pour chacune d'elles, identifié les registres d'entrée et de sortie, et avons décidé de garder la convention suivante pour mettre en œuvre nos fonctions : les registres `$aX` servent de registres d'entrée, et les `$vX` de registres de sortie. Après avoir réussi à afficher les différents éléments à l'écran, tels que les aliens, les bâtiments et le joueur, nous avons commencé à nous répartir les différentes tâches à effectuer : la gestion des vitesses pour chacun de ces éléments a été confiée à Lucas, tandis que l'implémentation de trois conditions de fin a été confiée à Gaëlle.

Pour la gestion des vitesses, un système de modulo (fonction auxiliaire `modulo` a été codé, pour gérer de manière conditionnelle l'activation (ou non) du tir, du déplacement des aliens et du déplacement du shooter. Du côté de l'implémentation des conditions de fin, la fonction `comptage_aliens` a été ajoutée, pour permettre de vérifier plus simplement si tous les aliens ont été tués.

La gestion de la quatrième condition de fin, qui détecte si les aliens ont atteint le sol, a été confiée à Lucas. Après une première version fonctionnelle, une fonction auxiliaire a été ajoutée afin d'affiner la condition de fin. La mort des aliens situés dans les couches inférieures va influencer la fonction précédente. Ainsi, le jeu ne se terminera pas si une ligne d'aliens vide atteint les bâtiments. Pendant ce temps, la mise en place d'un système de score, et de vies, a été réalisé par Gaëlle. Le score se calcule comme tel : +100 points par alien tué, -1 point par tir. Au niveau de l'implémentation des vies, une fonction teste si le joueur est bel et bien mort ou s'il lui reste une ou plusieurs vies. Au total, au début de la partie, il en possède 3.

A ce stade, les fonctions permettant au jeu de réagir comme le sujet le demande ont été réalisées. Nous avons donc pris la liberté d'ajouter quelques fonctionnalités supplémentaires à notre Mars Invaders. Par exemple, Lucas a programmé un système de cooldown qui agit entre chaque tirs, pour éviter l'effet "mitraillette" afin que le joueur ne gagne pas trop facilement. Des couleurs ont aussi été ajoutées au jeu, notamment sur chaque ligne d'aliens. Nous avons aussi ajouté un changement de couleur du shooter en fonction du nombre de vies restantes : moins il y en a, plus il sera rouge vif.

Enfin, afin d'afficher du texte à l'écran, une fonction va permettre d'afficher des caractères en fonction d'un code hexa représentant le caractère, et à partir de l'adresse à laquelle nous voulons afficher le caractère. Réalisée par Lucas, cette fonction a ensuite permis l'affichage d'une page de fin, indiquant si le joueur a gagné ou perdu, ainsi qu'une page de démarrage qui affiche le nom du jeu. C'est Gaëlle qui s'est occupée de la gestion de l'affichage du score sur l'écran final. Une fonction `afficher_score` a été implémentée. Son rôle est d'isoler chaque caractère composant le score final au moyens de soustractions successives, puis de les afficher à l'écran lorsqu'on a gagné ou perdu.