
Projet informatique Allegro



I) Présentation générale

1. Le projet
2. Bilan personnel

II) Le code

1. Structures de données
2. Graphe d'appel des sous-programmes
3. Particularités de notre jeu

III) Captures d'écran

1. Tableau des différents acteurs
2. Menu
3. Partie Niveau 1

I) Présentation générale

1. Le projet

Nous avons choisi de réaliser ce projet sur un thème spatial, avec pour acteur un vaisseau piloté par le joueur, des méchants sous plusieurs formes et des boss représentés par des monstres ou vaisseaux, le tout sur un fond défilant. Le jeu propose tout d'abord un menu qui donne accès à trois niveaux de difficulté différentes, ayant de même des décors différents.

Nous avons nommé notre version du R-TYPE, Orbitar.

2. Bilan personnel

- Ce projet m'a permis de me rendre compte de mes acquis en C et en programmation Allegro, mais aussi des limites de mes connaissances. Coder en vue d'un objectif bien concret (ici un jeu) a été une réelle motivation que je n'ai pas toujours en TP, et j'ai ainsi bien progressé. Il est parfois difficile de travailler en équipe, notamment lorsque celle-ci a été composée sans notre avis car la cohésion et l'entente ne sont pas toujours au rendez-vous ; cependant je trouve que notre trinôme a bien fonctionné et que chacun a pu apporter ses idées et ses connaissances pratiques au projet.

Eloïse Maurice

- A travers ce projet j'ai pu me rendre compte de ce qu'étaient mes limites à travers un projet concret tel que le jeu vidéo. De plus, m'étant principalement occupé des sprites et de leurs mouvements dans le jeu, j'ai eu l'occasion de perfectionner mon niveau en Allegro et en C. Je suis content d'avoir pu travailler avec Matthieu et Eloïse car notre vision de ce que nous voulions faire coïncidait parfaitement ce qui a permis d'éviter les éventuelles tensions et de créer un jeu qui rassemble les idées de chacun.

Tom Siennicki

- La réalisation d'un jeu vidéo a été pour moi une expérience ludique qui a permis d'appliquer mes connaissances sur un sujet bien concret. Je me suis rendu compte lors de mon travail que le langage C n'est pas adapté pour coder un jeu vidéo de par son organisation qui devient rapidement compliquée si on veut essayer de faire quelque chose un minimum complexe. Malgré cela, j'ai beaucoup aimé travailler avec Eloïse et Tom. En effet, nous avons réussi à satisfaire les envies de chacun ce qui a permis à tout le monde de s'investir et de s'amuser.

Matthieu Sajot

II) Le code

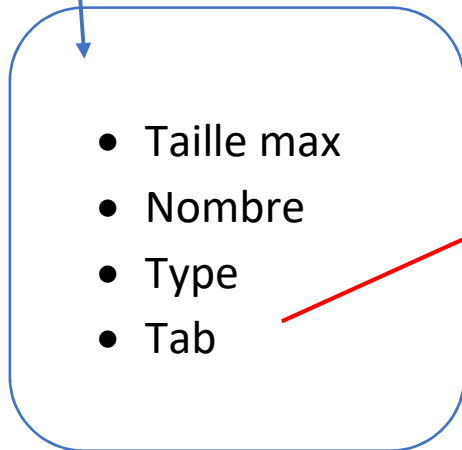
1. Structures de données

Pour répertorier les différents types de méchants et de tirs nous avons choisi de coder des sous-programmes dont les paramètres sont des structures de données réutilisables et modifiables aisément afin de les réadapter à chaque niveau :

Tout d'abord un tableau de pointeurs de structure *t_liste_acteur* de la taille du nombre de types différents d'acteurs ; la structure *t_liste_acteur* est composée de quatre champs : le nombre maximal d'acteurs (d'un type particulier) qu'il peut y avoir à l'écran à un instant donné (*int*), le nombre de ces acteurs qui sont utilisés (à l'écran) à un instant donné (*int*), le numéro du type d'acteur (*int*) et un tableau dynamique de pointeur sur structure *t_acteur* dont la taille maximale est le nombre maximal d'acteurs de la structure précédente, et le nombre de case allouées est le nombre d'acteurs utilisés de la structure précédente. La structure *t_acteur* comprend diverses champs *int* propres à chaque acteur de ce type : ses coordonnées, sa taille, sa vitesse, son état (mort ou vivant), ses points de vie, son temps de rechargement avant de pouvoir être réutilisé... qui varient en fonction de la nature de l'acteur (ennemi, missile, bonus).

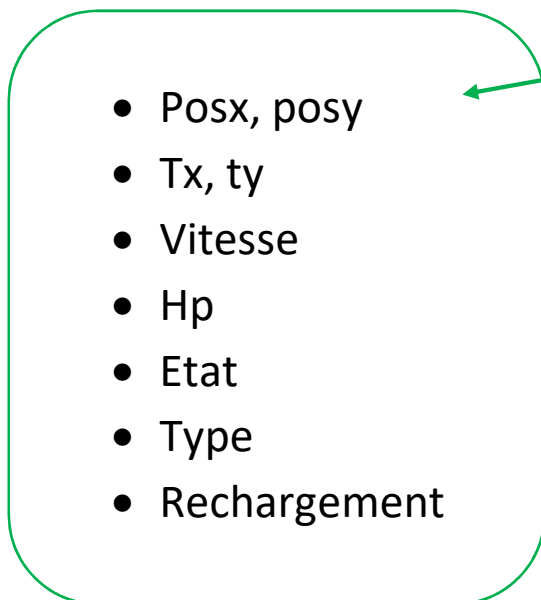
Type1			
-------	--	--	--

t_liste_acteur



Acteur1
Acteur2
Acteur3
Acteur4
Acteur5

t_acteur



2. Graphe d'appel

Initialisation :

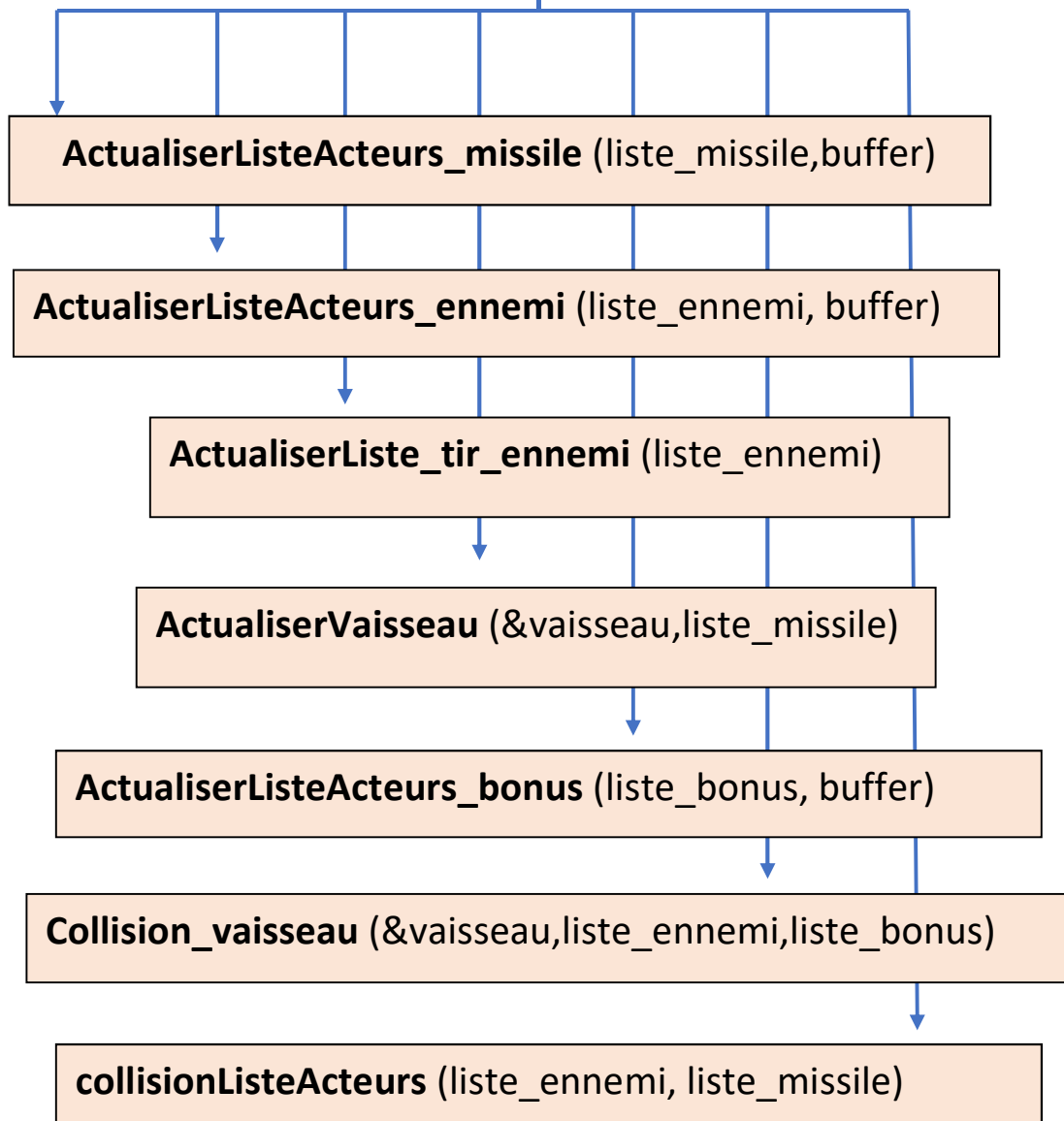
MAIN

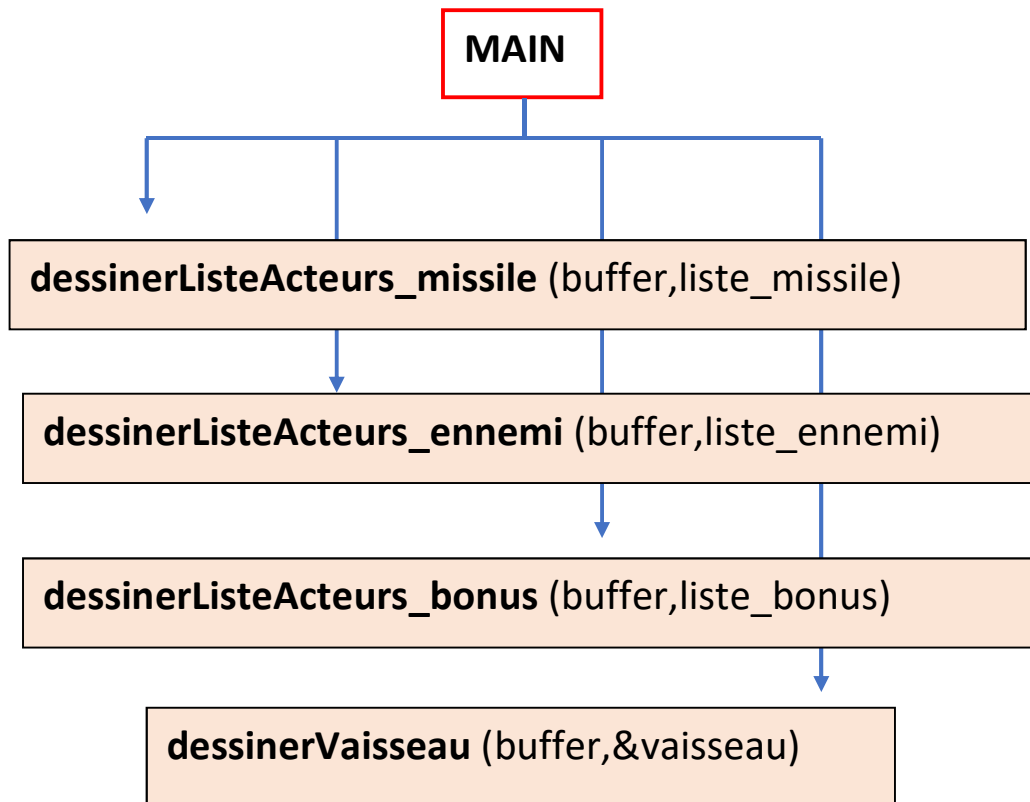
*Autant de fois qu'il y a de types de
missiles, d'ennemis et de bonus*

CreerListeActeur (nombre max de ces acteurs, numéro du type de ces acteurs)

Boucle de jeu interactive :

MAIN





La boucle interactive actualise à chaque tour les listes d'acteurs, d'ennemis et de missiles en fonction des actions effectuées par le joueur (au clavier) ; puis regarde s'il y a des collisions entre les différents acteurs en fonction de leur position actualisée, et enfin dessine le tout sur le buffer avant de l'afficher à l'écran.

3. Particularités de notre jeu

Notre version du jeu R-TYPE comprend trois niveaux de difficulté différente, avec des ennemis dont les apparences varient en fonction du niveau. Nous avons sonorisé les parties et ajoutés des bonus de vitesse et de vie dans les niveaux ; de plus le décor est interactif : dans le niveau 1, des météorites tombent aléatoirement du haut de l'écran et peuvent imputer les points de vie du vaisseau.

III) Captures d'écran

1. Listes des acteurs

TYPE	NUM		liste_ennemi	N
			liste_bonus	T
	1 listemissile[0]			
	2 listemissile[1]			
	3 listemissile[2]			
	4 liste_ennemi[0]			
	5 liste_ennemi[3]			
	6 liste_ennemi[1]			
	7 liste_ennemi[2]			
	8 liste_ennemi[4]			
	9 liste_ennemi[5]			
	10 liste_ennemi[6]			
	11 liste_bonus[0]			
	12 liste_bonus[1]			
	14 liste_bonus[2]			
	15 liste_bonus[3]			
	16 liste_bonus[4]			
	17 liste_ennemi[7]			
	18 liste_ennemi[8]			
	19 liste_ennemi[9]			
	20 liste_ennemi[10]			
	21 liste_ennemi[11]			
	22 liste_ennemi[12]			
	23 liste_ennemi[13]			

2. Menu



3. Partie Niveau 1

