

Journal de développement

Pour commencer, nous avons dû nous familiariser avec le langage C++ via des tutoriels sur internet puisque c'est la première fois que nous utilisons ce langage.

Pour travailler en groupe nous avons créé une interface GIT (un environnement de travail). Nous avons passé un peu de temps pour configurer notre répertoire et préparer notre environnement de travail.

Nous avons appris à séparer les classes en « .h » et « .cpp » ce qui facilite la clarté du code, mais aussi son organisation.

Au début du projet, lors de la création de la classe vecteur nous voulions pouvoir choisir entre travailler avec un vecteur à deux dimensions et un vecteur à trois dimensions. Cela compliquant la réalisation de notre travail et n'étant pas très explicite nous avons choisi de le retirer et de garder un vecteur en trois dimensions.

Pour la création de la classe particule, notre problème était de savoir où définir la position, la vitesse et l'accélération de l'objet particule dans la classe main ou particule. Finalement on a défini la position et la vitesse dans particule.

Dans la classe particule, nous avons eu un problème pour créer la fonction qui met à jour la position et la vitesse de la particule. En effet, à la base nous n'avions qu'un ensemble position et vitesse qui prenait les valeurs en (x, y, z) de la position et de la vitesse. Notre fonction ne fonctionnait pas et n'était pas clair, car on devait créer une boucle for pour effectuer les opérations sur chaque argument. Nous avons passé par trois paramètres en position (posx, posy, posz) et trois paramètres en vitesse (vitx, vity, vitz) ce qui nous permet de rendre le calcul plus simple et plus facile pour comprendre les erreurs.

Nous avons mis la gravité sur les trois axes x, y et z, alors qu'il fallait mettre la gravité seulement sur un axe (ici l'axe z).

Pour les calculs réalisés à chaque frame nous avons eu des difficultés. On a essayé d'adapter le programme au réel, mais cela ne fonctionnait pas, car « visual studio » mais du temps à calculer par rapport au temps réel, donc on pouvait perdre des espaces de temps. Donc on a pris comme temps = « $1/30 * \text{temps}$ » pour calculer en fonction du temps machine.

Nous avons perdu plusieurs heures à essayer d'intégrer « glut » une interface graphique à notre projet sans réussir à le faire fonctionner. Nous avons donc décidé de nous contenter d'un affichage console par manque de temps.