Plan de la soirée :

Qui fait quoi :

Maxime : Génération des terrains : Aléatoire, types de fonctions, type de terrain (boue, asphalte…), destruction, vent (variation, influence sur les tirs)

Nathan : Projectiles : Énergie vs physique, types de fonctions, position du curseur (direction en fonction de la position de la souris…) dommages des projectiles

Gabriel : IA, différences entre les niveaux IA : tirs, déplacement; …

Historique et démarche :

Choix du sujet :

On avait plusieurs choix de projets, dont celui-ci. On a évalué le potentiel de chaque idée : robot, simulateur spatial et, finalement, le jeu de tank. On a opté pour le dernier, car les autres proposaient soit un défi trop important (robot : apprivoiser la méthode de travail, apprivoiser les composantes et les mécanismes), soit trop facile (simulateur : beaucoup de constantes, les variations de trajectoires ne sont pas vraiment visible d’un point de vue humain sur une échelle de temps courte).

Deux questions s’imposaient lors du choix : Notre jeu est en temps réel ou en tour par tour? Notre jeu se jouera-t-il en multijoueur contre un autre joueur ou contre un IA? Après mûre réflexion (pas trop longue, car le temps s’écoule rapidement!), nous avons décidé de faire un jeu en temps réel et contre une intelligence artificielle.

Erreurs et solutions: L’implémentation des tirs énergies ont causé beaucoup de problèmes. Plusieurs méthodes ont été utilisé et plusieurs heures ont été perdu dans sa création puisque créer une ligne graphique et l’effacer au même moment causait plusieurs erreurs difficiles à régler. La solution a été de créer initialement plusieurs petites parties du tir et les afficher de manière à donner un effet de mouvement. La création du terrain a été difficile et plusieurs erreurs d’inattention a été commis puisque le point (0,0) est en haut à gauche et non en bas à gauche de l’écran. Les erreurs ont été solutionnés par une bonne attention et modélisation intuitive du terrain et son implémentation inverse dans le code. Le mouvement du tank a causé plusieurs erreurs. Premièrement, le tank sautillait sur le terrain et ne montait pas les pentes, donc nous avons modifié les algorithmes de collision du tank sur le terrain. Ensuite, le tank n’avait jamais la bonne orientation. À la place d’utiliser un algorithme qui détecte la collision sur le point central du tank, nous avons utilisés deux points de collisions aux extrémités du tank, de cette façon, la gravité agissant sur les deux points, l’orientation du tank se dessinait avec la tangente entre les deux points. Nous avons essayé de développer un algorithme parfait de tir de projectile pour notre plus grande difficulté d’IA. Le problème était plus complexe qu’il n’y paraissait. Premièrement, la démarche mathématique comprenait de déterminer la forme close de la relation de récurrence du mouvement. Puisque cette forme close nous donnait une formule paramétrés d’une parabole, nous devions unir les fonctions X(t) et Y(t) en Y(X). Ensuite, dans notre situation, avec le programme, nous avions accès à deux points qui caractérise la fonction soit la position initiale et la position finale. Un troisième pouvait être déterminé par un algorithme de recherche du point d’intersection à la limite de collision avec le terrain pour assurer une trajectoire parfaite. Le plan était d’utiliser les 3 points dans la fonction pour faire une résolution matricielle pour isoler le vecteur de tir. Par manque de temps et de problème algébriques, l’idée n’as pas pu être implémenté.

Matériel : Ordinateur portable de Gabriel à brancher avec le tableau interactif (VGA ou HDMI). L’affiche et de beaux habits.

Scénario :  
Expert :

* Génération des terrains : Aléatoire, Sinus, Sinus cube, sin(x)\*(x+cos(x))
* types de fonctions, sin, cosec, abs, rationnelle, tan; et les problèmes de développement.
* type de terrain (boue, asphalte, gazon),
* destruction du terrain via Pythagore et problèmes.
* vent (variation, influence sur les tirs)
* Projectiles : Énergie vs physique, position du curseur (direction en fonction de la position de la souris…) dommages des projectiles (hit box)

IA, différences entre les niveaux IA : tirs, déplacement; …

Novice :

* Génération du terrain Sinus
* Tir de fonction sinus
* Type de terrain
* destruction du terrain
* Projectiles : Énergie vs physique, position du curseur (direction en fonction de la position de la souris…) dommages des projectiles

Sources :

Musiques:Toccata et fugues en D mineur (Jean-Sébastien Bach),(<https://www.youtube.com/watch?v=ipzR9bhei_o>)

La Chevauchée des Walkyries (Richard Wagner), (<https://www.youtube.com/watch?v=KMTRqAgLw04>)

With Gun and Crucifix (Denny Schneidemesser), (<https://www.youtube.com/watch?v=aV3kPtYm6Qw>)

Thème musical du jeu Battlefield 1 (DICE, Electronic Arts),(<https://www.youtube.com/watch?v=SV0IF4XVBlE>)

Ghost Division (Sabaton),de l'album The Art of War

Les autres musiques de menu sont prises du site de musiques gratuites [www.bensound.com](http://www.bensound.com)

Photos:  
Arrière-plan du menu principal: Image réalisée parkaratastamer,(<http://wallpapersafari.com/w/8nDT4Y/>)

Arrière-plan du menu des options: (<https://giphy.com/gifs/tank-bridge-crossing-P8XNUbh3Wuwa4>)

Tous les droits des musiques et des photos vont à leur(s) créateur(s) respectif(s).

Ce qui distingue notre projet des autres jeu de tanks tel que shellshock© est l’unique variété de tirs à notre possession lors du temps de jeu, et surtout, le jeu en continue qui s’écarte du style tour-par-tour de ce genre de jeu.

Présentation de concepts scientifiques :

* Beaucoup d’application de formules trigonométriques,
* Utilisation de formules paramétrées du seconde degré (MRU en X, MRUA en Y) en formules de récurrences.
* Résistance du vent
* Rebond de surfaces variables