



Nom de votre projet	JOmetry
Membre de l'équipe n°1 (prénom/nom)	Mickael Mao
Membre de l'équipe n°2 (prénom/nom)	Solal Pivron Djeddi
Membre de l'équipe n°3 (prénom/nom)	Martin Vidal Oger
Membre de l'équipe n°4 (prénom/nom)	
Membre de l'équipe n°5 (prénom/nom)	
Niveau d'étude (première ou terminale)	Terminale
Établissement scolaire	Lycée Louis-le-Grand
Responsable du dépôt (professeur de NSI)	Loïc Josse

1 / PRÉSENTATION GÉNÉRALE

Pouvez-vous présenter en quelques mots votre projet ?

Comment est né ce projet ? Quelle était la problématique de départ ?

Quels sont les objectifs ? À quels besoins répondez-vous ?

Notre projet est un logiciel de géométrie compatible avec la géométrie projective. Par un beau jour de printemps, alors que des élèves discutaient avec un sympathique problème de géométrie, ils se sont rendu compte qu'ils ne pouvaient pas tracer de cubiques isoptiques sur Geogebra, le logiciel qu'ils utilisaient. En effet, réaliser cette courbe nécessite de pouvoir tracer des cubiques, ce qui n'est pas rendu automatique sur Geogebra. Dès lors, il s'agissait de construire un nouvel environnement apte à gérer ces objets, ce qui a été rendu possible par l'implémentation d'une application fondée cette fois-ci sur le plan projective complexe. Non contents de s'arrêter là, nous avons continué et implémenté toutes les fonctionnalités de géométrie dont nous avons besoin pour traiter de problèmes complexes ce qui a abouti à ce logiciel que nous vous soumettons. Actuellement, il se distingue donc par son aptitude à conceptualiser des courbes algébriques de degré plus petit que 6, ainsi que par sa compréhension des points cycliques, lesquels sont à la base du plan projectif complexe.

2 / ORGANISATION DU TRAVAIL

Vous veillerez au bon équilibre des différentes tâches dans le groupe !

Chaque membre de l'équipe doit obligatoirement réaliser un aspect technique du projet réalisé (hors design, gestion de projet, rédaction, montage vidéo).

Pouvez-vous présenter chaque membre de l'équipe et préciser son rôle dans ce projet ?

Pourquoi cette organisation du travail et comment avez-vous réparti les tâches ?

Combien de temps avez-vous passé sur le projet ?

Quels sont les outils et/ou les logiciels utilisés pour la communication et le partage du code ?

Solal : Passionné de mathématiques, il a posé les bases du système de calcul dans Engine.py et s'est occupé de réaliser de nombreuses fonctions liées aux calculs projectifs.

Mickael : Il a réalisé tout ce qui concerne l'aide pour l'utilisateur : la fenêtre d'aide, les tooltips ainsi que les fichiers

Martin : Expert en tkinter, c'est à lui que l'on doit toute l'interface graphique du programme. De plus, il a traqué toutes les incohérences laissées par ses deux compères, et a reconstruit toute la nomenclature du programme.

Cette répartition du travail s'est imposée d'elle-même face aux compétences et habitudes de chacun. Par ailleurs, le bagage mathématique nécessaire à la

compréhension du cœur du programme étant très avancé, il n'était pas possible pour tout le monde de gérer cette partie.

Il est compliqué d'estimer le temps passé sur ce projet, mais il ne semble pas absurde qu'à nous trois, il s'agisse d'un travail cumulé d'environ 500 heures puisque nous n'en avons pas commencé la conception cette année.

Le partage du code se faisait essentiellement par l'ENT et whatsapp, jusqu'à la mise au point très récente du github auquel vous avez accès.

3 / ÉTAPES DU PROJET

Présenter les différentes étapes du projet (de l'idée jusqu'à la finalisation du projet)

Initialement, il s'agissait de construire le monde sur lequel se base notre projet : le plan projectif complexe. Une fois cela fait, nous avons de premiers fondements solides à partir desquels développer le reste de notre géométrie. Nous avons dans un premier temps rendu possible le tracer de courbes algébriques de degré plus petit que 10.

Ensuite sont venues des fonctions plus élémentaires telles que les droites, les milieux, les segments, les perpendiculaires, etc...

Muni d'une multitude d'objets, il nous fallait maintenant être en mesure de les intersecter, d'où l'introduction des bases de groebner dans le projet, nous permettant d'intersecter des courbes algébriques quelconques très efficacement.

Par suite, nous nous sommes concentrés sur l'implémentation des transformations du plan, basées sur des matrices 3×3 : rotation, translation, homothétie, symétrie... Ainsi que bien sûr inversion.

Après un bref détour par la dualisation permettant le tracer de tangentes communes à des courbes, nous étions dotés d'un attirail solide de fonctionnalités, essentiellement toutes celles dont nous disposons aujourd'hui. Nous avons donc décidé de remédier à l'interface graphique largement manquante, ainsi qu'à l'ajout de fonctionnalités intéressantes pour l'utilisateur mais davantage reliées au programme en tant que tel : travailler en simultanée, sauvegarder un projet, en ouvrir un, undo, redo...

4 / FONCTIONNEMENT ET OPÉRATIONNALITÉ

Pouvez-vous présenter l'état d'avancement du projet au moment du dépôt ? (ce qui est terminé, en cours de réalisation, reste à faire)

Quelles approches avez-vous mis en œuvre pour vérifier l'absence de bugs et garantir une facilité d'utilisation de votre projet ?

Quelles sont les difficultés rencontrées et les solutions apportées ?

Actuellement, nous sommes en mesure de réaliser une immense majorité des constructions géométriques possibles : interpolation de droites, cercles, coniques, perpendiculaires, parallèles, milieux, rotation, tangentes, inversion...

Il nous reste néanmoins à implémenter les lieux de points, permettant de répondre à des questions en apparence naturelle de la forme « Soit A et B deux points du plan. Quel est l'ensemble des points X du plan tel que l'angle $AXB=90^\circ$ ». En l'occurrence il s'agit du cercle de diamètre [AB]. Cette fonctionnalité est l'une des motivations principales du projet, mais est encore hors d'atteinte pour le moment et nécessite des outils plus avancés basés notamment sur les anneaux de Chow de ce que nous en avons compris. Néanmoins, avec quelques mois de travail supplémentaire, nous pensons être en mesure d'y remédier.

Nous voulons aussi implémenter le moving point, mais c'est malheureusement trop compliqué à expliquer simplement.

Finalement, même si par une grande série de tests effectuées par notre professeur, nos camarades de classe, ainsi que nous-mêmes, nous avons réussi à éradiquer une grande majorité de bugs, certain dont nous sommes conscients subsistent encore, et nécessiteraient de revoir entièrement notre conceptualisation du calcul de coordonnées d'objet, notamment après des transformations du plan.

Les principales difficultés que nous avons rencontrées sont essentiellement dues à la liaison brutale entre le modèle mathématique, beau simple et efficace, et l'aspect pratique python. En effet, python peut se révéler très lent pour calculer le déterminant de matrices, et est naturellement rapidement apte aux erreurs de calcul face à des nombres très grands de l'ordre de 10^{120} par exemple, auxquels nous sommes rapidement confrontés. Nous n'avons pas pu résoudre cela autrement qu'en bidouillant quelques divisions par des puissances de 10 lorsque les nombres se révélaient trop grand. Mais naturellement cela ne marche pas toujours.

De plus, dans l'affichage de cercles par exemple, il est compliqué de distinguer un point réel d'un point calculé par python comme racine d'un polynôme, dont la partie imaginaire est de l'ordre de 10^{-20} . Ce dernier problème a pu être résolu par l'idée d'intersecter systématiquement nos courbes avec le plan réel, de sorte à ne pas laisser de nombres complexes dans leurs coordonnées.

5 / OUVERTURE

Quelles sont les nouvelles fonctionnalités à moyen terme ? Avez-vous des idées d'amélioration de votre projet ?

Pourriez-vous apporter une analyse critique de votre projet ? Si c'était à refaire, que changeriez-vous dans votre organisation, les fonctionnalités du projet et les choix techniques ?

Quelles compétences/appétences/connaissances avez-vous développées grâce à ce concours ?

En quoi votre projet favorise-t-il l'inclusion ?

Nous aimerions implémenter l'étude automatique de notre figure : étant donné une figure ainsi que des points fixes renseignés au programme, il s'agirait de reporter à l'utilisateur toute quantité invariante. Par exemple étant donné deux points fixes A et B, ainsi qu'un point C sur un cercle fixé passant par A et B, le programme nous informerait le résultat connu qu'est l'invariance de l'angle orienté ACB. Cela permettrait aussi de trouver automatiquement des concourances de droites et de les montrer « algébriquement ».

Accessoirement, comme vous pourrez vous en rendre compte avec le fichier philosophie, nous étions en train d'implémenter la conceptualisation de points en tant qu'indéterminées, et non pas coordonnées fixées. Cela permettrait de transformer tout calcul de coordonnées par une simple application numérique, bien plus rapide.

Finalement, nous réfléchissons à comment tracer de manière plus efficace des courbes algébriques de degré plus grand que 3. Nous ne disposons en effet pas toujours d'une paramétrisation gentille, accessible pour les degrés plus petits que 2. Il semblerait que ce soit un sujet connu, à propos duquel se trouve nombre de ressources en ligne, mais nous n'avons pas encore réussi à en implémenter une solution efficace parmi celles proposées. Il faudrait aussi pouvoir tout retracer si l'on bouge le plan, ce qui n'est pas encore implémenté.

Si c'était à refaire, nous aurions préféré laisser la partie calcul à un programme réalisé en C++ : python est encore trop lent. De plus, nous aurions aimé concevoir clairement la structure du programme avant de le programmer, ce dont nous nous sommes passés, conduisant à un code relativement brouillon, qui a souvent été entièrement remodifié. Encore aujourd'hui, il est long et fatigant de débayer notre programme tant toutes les fonctions s'entremêlent les unes dans les autres, ce que nous aurions sans doute pu éviter.

Ce projet nous a beaucoup appris à la fois sur tkinter et sur la programmation orientée objet. Forcés d'appriover ces concepts dont nous étions plutôt éloignés, nous avons pu en approfondir notre compréhension grâce à ce projet. De plus, chacun ayant une manière assez unique de coder, la relecture du travail de nos camarades s'est révélée très riche et nous a beaucoup appris, notamment en matière d'astuces python.

Finalement, ce projet promeut l'inclusion puisque dans notre monde, les droites, cercles, coniques et courbes algébriques quelconques ne sont aucunement séparés : tous sont traités de la même manière, comme « Créatures », et ce malgré leurs différences apparentes. Ce n'est qu'avec l'affichage, que nous sommes forcés de les distinguer.