# Sprint Backlog n°1

1. **Rappel des attentes client**

* Etablir un algorithme de sélection de la meilleure fonction booléenne représentant la matrice X ainsi que son observation.
* Le critère de sélection se basera sur le calcul de la fitness de la fonction, mais également du nombre imposé par le client de génération filles et du seuil de tolérance donné.
* Seuls des opérations de type AND, OR et NOT peuvent être utilisés.
* Il n’est pas nécessaire que toutes les variables de la matrice X (x1, x2…) apparaissent dans la fonction finale.
* Veiller, si possible, à ne pas être gourmand en mémoire et en complexité.

1. **Planning prévisionnel**

* 18/05 : Feedback de nos avancées à la classe + sprint 1
* 25/05 : Feedback du premier sprint, ajustement des attentes clients, prévision du deuxième sprint
* 08/06 : Eventuellement fin du second sprint, feedback, peaufinage et envoi du projet

1. **Détail des deux sprints (9h chacun)**

|  |  |
| --- | --- |
| Sprint 1 | Sprint 2 |
| * Import des données * Calcul de fitness * Test fuites mémoires * Classes nœuds (variable & opération) * Pseudocode mutation dirigée * Fonctions ajout/suppression… * Code pour créer une fille * Comparaison/Sélection des filles * Fonction de parcours des arbres * Documentation/Tests unitaires * Makefile | * Python : appel du code * Python : récupération des données * Python : affichage des données * Python : simplification de la formule * Préparation soutenance ? * Tests sur les données * Export des données * Implémenter mutation dirigée * Tests globaux * … |

1. **Ordre chronologique et répartition**

* Yacout  s’occupera de l’import des fichiers de données sous la forme csv. Le but est de créer une ou plusieurs fonctions dans la classe main capable d’extraire le fichier en question et de le rendre sous la forme d’un tableau. Ainsi, son utilisation sera bien plus souple.

Elle travaillera également sur le Makefile favorisant la compilation de toutes les classes.

* Miléna et Nicolas s’attaqueront à la classe « arbre ». En plus d’implémenter les constructeurs des différents arbres mais aussi le destructeur, ils devront réfléchir à la génération d’arbre filles tout en gardant une fitness meilleure que la génération précédente. Pour cela, ils devront créer de nombreuses fonctions et être clairs sur la liste d’attributs de la classe. Ils implémenteront notamment une fonction qui calcule la fitness, des fonctions permettant de générer les différentes mutations (ajout, suppression, modification) au sein des générations, mais également d’autres méthodes comme par exemple celle qui génère un arbre de façon aléatoire. Cette dernière méthode peut par exemple être utile au tout début de l’algorithme, en générant 5 arbres qui se proposent solution du problème.
* Quant à Vincent et Charles, ils travailleront sur la classe « nœud ». En particulier, ils utiliseront les notions d’héritage afin de différencier les nœuds qui sont de type variable de ceux qui sont opérateurs. Ils élaboreront également les différents constructeurs (nœud crée à partir d’une modification de type NOT, nœud pour AND et OR…) mais également le destructeur.
* Evidemment, tous les membres du projet sont impliqués dans chacune des parties du projet. Cette répartition nous permet surtout de poser des bases et de commencer à travailler plus efficacement.