

## **RAPPORT DE PROJET DE COMPILATION:**

Dans ce projet, nous avons construit un interpréteur en OCaml. Pour y parvenir, nous avons implémenté un lexer, un parseur, un vérificateur de type et un interpréteur. Ce projet est composé de cinq parties: l'arithmétique, les variables, les fonctions, les structures et la récursion.

### **Partie Arithmétique:**

Nous avons implémenté les opérations arithmétiques de base (addition, soustraction, multiplication, division et booléennes). Nous avons rencontré des difficultés pour comprendre la structure du code et pour l'implémenter en OCaml, mais nous avons finalement réussi à surmonter ces obstacles.

### **Partie Variables:**

Nous avons rencontré des difficultés similaires à celles de la première partie, mais nous avons réussi à les surmonter et à mettre en place cette partie correctement.

### **Partie Fonctions:**

Les principales difficultés que nous avons rencontrées étaient liées à la manipulation de la mémoire et à l'appel des fonctions. Nous avons réalisé des schémas de la table de mémoire et de l'environnement pour mieux comprendre comment stocker et gérer les données. Nous avons réussi à mettre en place cette partie correctement.

### **Partie Structures:**

Nous avons réussi à reconnaître une définition de structure, mais le reste de cette partie est encore en commentaire. Nous avons eu du mal à implémenter une fonction permettant l'accès à un champ "x" donné d'une variable de type structure. De plus, nous avons rencontré des difficultés pour accéder à un champ de structure et pour identifier ce que représentent les expressions récupérées lors de l'analyse grammaticale afin de les utiliser pour le typage et l'interprétation.

### **Partie Récursion:**

La partie lexer, parseur et vérificateur de type fonctionne correctement, mais il y a un bug dans l'interprétation. Nous avons constaté qu'une fonction récursive renvoie un pointeur au lieu d'une valeur, ce qui peut être dû à une mauvaise implémentation de l'opérateur Fix ou d'un problème de mise à jour de l'environnement dans l'opérateur App.

### **Extension:**

Nous avons tenté d'implémenter les types énumérés dans ce projet, mais nous avons rencontré des difficultés pour comprendre leur fonctionnement et pour les implémenter de manière adéquate dans notre interpréteur. Malgré nos efforts, nous n'avons pas réussi à mettre en place cette extension correctement.

**Tests:**

Pour vérifier que chaque partie fonctionnait correctement, nous avons ajouté différents cas de tests dans le dossier "tests" afin de vérifier que les commandes agissaient comme prévu dans leur environnement.