

## PLD-COMPILATEUR - GESTION DE PROJET

---

Hexanôme 4244

ALAMI Meryem

AL ZAHABI Hanaa

BELAHCEN Basma

CHELLAOUI Adam

GREVAUD Paul

GUILLEVIC Marie

M'BARECK Aichetou

PREVOT Jade

WAQIF Salma

10 avril 2022



## Table des matières

<b>1</b>	<b>Présentation de l'équipe</b>	<b>2</b>
<b>2</b>	<b>Gestion de projet</b>	<b>2</b>
2.1	Déroulé première partie . . . . .	2
2.1.1	Première séance . . . . .	2
2.1.2	Deuxième séance . . . . .	2
2.1.3	Troisième séance . . . . .	3
2.1.4	Quatrième séance . . . . .	3
2.2	Déroulé deuxième partie . . . . .	3
2.3	Outils utilisés . . . . .	4

## 1 Présentation de l'équipe

Notre équipe se compose de 9 personnes :

- GUILLEVIC Marie : Chef de projet
- ALAMI Meryem
- AL ZAHABI Hanaa
- BELAHCEN Basma
- CHELLAOUI Adam
- GREVAUD Paul
- M'BARECK Aichetou
- PREVOT Jade
- WAQIF Salma

## 2 Gestion de projet

Pour la gestion de projet, nous avons décidé de travailler par cycle entre chaque séance de cours.

### 2.1 Déroulé première partie

Pour la première partie du projet, il fallait se concentrer sur la réalisation d'un compilateur minimal qui se compose d'un analyseur syntaxique (parser) et d'un générateur de code. Pour que chacun maîtrise bien les bases, nous avons décidé de travailler en groupe. Étant 9 dans l'hexanôme, nous nous sommes divisé en 2 groupes travaillant en parallèle sur les mêmes tâches.

#### 2.1.1 Première séance

Pendant la première séance, nous avons commencé par installer antlr. Nous avons par la suite procédé à la prise en main du squelette de code fourni.

#### 2.1.2 Deuxième séance

Pendant la deuxième séance, chaque groupe de son côté a travaillé sur les déclarations et affectations de variables. Nous avons donc réussi à l'issue de cette séance à déclarer plusieurs variables sur une même ligne. De plus, nous sommes également parvenus à affecter à chacune des variables déclarées une valeur (une constante ou une autre variable déclarée).

### 2.1.3 Troisième séance

Pendant la troisième séance, nous avons travaillé sur les expressions arithmétiques ainsi que sur le moins unaire. À l'issue de cette séance, nous avons réussi à réaliser les calculs nécessaires et donc générer le code assembleur pour les expressions arithmétiques de base (additions, soustractions, multiplications et divisions). De plus, nous avons veillé à respecter l'ordre des opérations (la multiplication et la division avant l'addition et la soustraction).

### 2.1.4 Quatrième séance

Pendant la quatrième séance, nous nous sommes consacré en priorité à la finition des tâches précédentes et à la rédaction des livrables de mi-parcours. Nous avons aussi entamé la réalisation du compilateur final qui utilise l'IR. Pour pouvoir travailler le plus efficacement possible, nous nous sommes répartis les tâches en binôme ou trinôme.

- Une équipe sur l'implémentation de IRInstr
- Une équipe sur l'implémentation de BasicBlock
- Une équipe sur l'implémentation de CFG

## 2.2 Déroulé deuxième partie

Pour la deuxième partie, nous avons pour objectif de réaliser notre compilateur final. Nous nous sommes réparti les tâches en binômes ou en trinômes.

Sur les 2 premières séances, nous avons travaillé sur 4 tâches principales réparties en 3 binômes et 1 trinôme :

- Ajout du type char et de l'inférence de type
- Mise en place de l'enregistrement d'activation, de l'ABI et des appels de fonction avec vérifications statiques.
- Compiler le if ... else.
- Compiler l'affectation à une lvalue quelconque et compiler des tableaux.

Pour la 3ème séance, nous avons ajoutés quelques fonctionnalités :

- Gestion du return n'importe où.
- Compiler les boucles while.

La dernière a été consacrée à la présentation du compilateur.

### 2.3 Outils utilisés

Nous utilisons plusieurs outils pour réaliser le projet :

- Notion pour la gestion de projet.
- Github pour le versionning.
- Visual Studio Code pour le développement.
- Overleaf pour la rédaction des rendus.