1. **Introduction**
2. **Contexte général et description de la problématique**
   1. **Problématique**

Comment modéliser une population de lapins dont le nombre change au fil du temps en fonction de mécanismes internes à savoir : la maturité des individus, le nombre de naissances et le décès tout en négligeant les influences extérieures.

* 1. **Lois**

Pour concevoir la solution, nous avons besoin de définir les lois de probabilité qui régissent l’évolution de notre population :

* *Number of litters:*

Chaque lapine donne naissance de 4 à 8 fois par an, sachant que la possibilité d’avoir 5, 6 et 7 est supérieur aux autres. La loi de probabilité définissant le nombre de litters est donc donnée par :

* + **25% :** Pour 5, 6 et 7
  + **12% :** Pour 4
  + **13% :** Pour 8
* *Number of babies per litter:*

Dans chaque litter on peut avoir de 3 jusqu’à 6 bébés lapins, nous allons donc utiliser une loi uniforme d’une valeur de **25%** pour chaque cas.

* *Death:*

Pour le nombre de décès, nous avons opté pour les probabilités suivantes :

* + Pour les bébés lapins (< 1 an) : **65%**
  + Pour les adultes (entre 1 et 10 ans) : **40%**
  + Après 10 ans, le taux de décès s’incrémente de **10%** chaque année
  + Les lapins ayant plus que 15 ans vont donc tous mourir
* **Remarque**: Les deux premières lois seront modifiées durant la phase de test afin d’illustrer leurs impacts sur l’évolution de la population.

1. **Conception et modélisation**
   1. **Structure de stockage**

Pour organiser notre travail, nous avons opté pour les 2 structure de données suivante :

* *\_ageCategory* :

Cette structure contient les informations nécessaires sur chaque catégorie d’âge, à savoir le nombre de lapins et le pourcentage de femelle.

* *\_statistic* :

Cette structure est utilisée pour stocker les statistiques de chaque année à savoir le nombre total de décès, de naissance, de la population et enfin et le nombre total de femelle. Ces statistiques seront utilisées pour générer des graphes afin de constater l’impact d’initialisation de la population et de choix de lois de probabilités sur le résultat final.

* 1. **Description des fonctions**

Les fonctions principales dans notre code sont les suivantes :

* + **void Death (vector<\_ageCategory> & population, \_statistic & statistic):**

Cette fonction permet de diminuer le nombre des individus dans chaque catégorie d’âge tel que chaque catégorie a un taux de survie qu’on l’utilise pour calculer le nombre de décès dans celle-ci. Cette fonction est appelée une fois par un, donc la somme de nombre de décès dans toutes les catégories est stockée dans total\_deaths de la structure statistics.

* + **void Birth (vector<\_ageCategory> & population, \_statistic & statistic)**

//salma

* + **void Update\_total\_individuals (vector<\_ageCategory> & population, \_statistic & statistic)**

Tout d’abord, dans cette fonction, on modifie total\_deaths de l’année en ajoutant le nombre des individus qui existe dans la catégorie d’âge 15 car ils seront tous morts. Ensuite les individus de chaque catégorie d’âge passeront à la catégorie suivante car une année été passée dans l’âge sera incrémenté et la première catégorie reçoit le nombre de naissance de l’année courante. Enfin on calcule le nombre totale de la population qui sera stocké dans total\_population de la structure statistics

* + **void Update\_total\_female (vector<\_ageCategory> & population, \_statistic & statistic)**

//salma

* 1. **Structure de programme**

1. **Réalisation et interprétation** 
   1. **Graphe 1 :**
   2. **Graphe 2 ..**
2. **Conclusion**