# Polytech4A Info Apprentissage
# Université Claude Bernard Lyon 1

## Projet crypto

A rendre avant le 09/06/2023

- *Le projet pourra se faire en binôme.*
- *Les réponses aux questions seront rédigées séparément du code dans un fichier au format pdf.*
- *Votre code + pdf est à déposer sur Tomuss et à envoyer par mail à l'adresse suivante :* `gerald.gavin@univ-lyon1.fr`*. Vous enverrez une archive portant le nom* `NOM.tar.gz` *(ou zip) contenant un seul fichier python. Vous pourrez utiliser le code que vous avez développé au cours de l'UE. Vous importerez un minimum de bibliothèques*
- **La note tiendra compte du respect des consignes.**

Après de nombreuses années de mariage cryptographique, Bob a eu un $n^{ieme}$ comportement malveillant vis à vis d'Alice lors d'un protocole houleux. Une mesure d'éloignement à été prononcée contre lui. Pour garantir ceci, il est affublé d'un bracelet numérique connecté (muni d'une petite capacité de calcul) et géolocalisé. Le "bracelet" connait sa position, i.e. ses coordonnées $(x_B, y_B) \in \mathbb{Z}^2$ dans un repère orthonormé arbitraire dont l'unité est le mètre. Les coordonnées d'Alice seront notées $(x_A, y_A) \in \mathbb{Z}^2$

**On supposera dans tout le TP que Bob (via son bracelet electronique) est honnête ou malhonnête passif signifiant qu'il respecte le protocole, e.g. le protocole est implémenté en hard sur son bracelet.**

Nous supposerons dans tout le TP qu'Alice a généré un couple de clés $(pk, sk) \longleftarrow \mathsf{Paillier.KeyGen}(\lambda)$ et que Bob (son bracelet) connait $pk$. Pour simplifier les notations, $[x]_{pk}$ (ou simplement $[x]$ lorsqu'il n'y aura aucune ambiguité sur la clé publique) désignera une encryption d'une valeur $x$ avec la clé publique $pk$.

Q1 Il s'agira d'établir un protocole DistanceBob entre Alice et le bracelet, permettant à Alice de connaitre la distance $d_{AB} = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$ à laquelle Bob se trouve. Voici une version non-détaillée de ce protocole qu'il s'agira de détailler.

---

**DistanceBob**

---

1. **Alice** envoie $[x_A]$ et $[y_A]$ à Bob
2. **Bob** Envoie $[x_B^2 + y_B^2 - 2(x_A x_B + y_A y_B)]$ à Alice
3. **Alice** retourne $d_{AB}$

---

Q2 Implémenter ce protocole.

Q3 Analyser la sécurité de ce protocole.

Q4 Proposer une variante DistanceBob100 du protocole précédent permettant à Alice de connaitre seulement si $d_{AB} \leq 100$ ou non (et non plus $d_{AB}$).

Q5 Analyser la sécurité de DistanceBob100.

Q6 Implémenter DistanceBob100.

Q7 Proposer (sans l'implémenter)/analyser une variante LocalisationBob100 permettant à Alice de connaitre $(x_B, y_B)$ si et seulement si $d_{AB} \leq 100$.

**Indice.** On pourra utiliser le fait que $100/n$ est négligeable, i.e. $100/n \leq 2^{-\lambda}$.