Programmation répartie **Votre nom:** _____ **2020 – 2021**

DS n°1 *Devoir*

Exercice 1. Messagerie

On souhaite modéliser un système de messagerie.

Soit un serveur de messagerie, sur lequel il est possible d'envoyer et de récupérer des messages. Sur ce serveur chaque utilisateur a une boite de réception, **limitée à 10 messages**. Chaque utilisateur est identifié par son nom.

Le serveur met à disposition les méthodes suivantes :

- Pour envoyer les messages :
  `sendto(String destinataire, String message)`.
  Si la boite de réception du destinataire n'est pas pleine, le message est ajouté. Si elle est pleine, on devra attendre qu'il y ait de la place, le thread est donc bloqué.
- Pour la réception des messages :
  `String recv(String user)`
  Cette méthode récupère un seul message, et le supprime de la boite de réception. Si la boite est vide, le thread est bloqué, en attendant la réception d'un message.

Les clients utilisent ces méthodes pour relever leur courrier, ou envoyer un message.

Faites en sorte que les boites de réception des utilisateurs soient créées à l'initialisation. Vous simulerez l'exécution, avec des envois et des réceptions. Pour simuler que la réception prend du temps, ajoutez une pause (5s) entre deux récupérations. Dans votre exemple faites en sorte que la boite d'un utilisateur arrive à saturation.

Voici un exemple d'affichage :

```
Bob récupère le message : "Bonjour Bob"
Alice envoie le message à Carl : "Au revoir Carl"
La boite de Bob est vide
Alice attend que la boite de David ne soit plus pleine
```

1.1 Réalisez ce programme en Java. Programmez le serveur, le client, et les autres classes nécessaires. Dans un premier temps faites un client simple, qui consulte sa boite, puis envoie des messages. Plusieurs client sont actifs simultanément.

Dans un second temps, faites en sorte que pour un client la consultation et l'envoi soient fait en parallèle.

Déposez sur Célène votre archive portant votre nom et prénom.