



Random avec bornes

1 Descriptif

L'objectif de ce défi est d'implémenter une méthode recevant deux nombres binaires : min et max en paramètre (avec $\text{min} < \text{max}$) et devant générer un nombre binaire aléatoire supérieur ou égal à min et strictement inférieur à max.

2 Protocole

1. Une fois la connexion établie, le serveur commence par envoyer un premier message annonçant le début du défi :

-- Début du défi : Random avec bornes --

Ce message n'attend pas de réponse.

2. Le serveur envoie ensuite une série de nombres binaires (de taille aléatoire) deux par deux.
3. Pour chaque paire de nombres binaires, le serveur doit recevoir en retour un nombre binaire (sous forme binaire) supérieur au premier et strictement inférieur au deuxième.
4. Après chaque réponse, le serveur enverra un message commençant par "OK" ou "NOK" suivant si la réponse est correcte ou non.
5. A la fin du défi, le serveur enverra un message indiquant "Défi validé" ou "Défi échoué!". Aucune réponse n'est attendue.
6. Le serveur terminera la communication par le message "FIN", votre client devra alors fermer la socket. Aucune réponse n'est attendue.

3 Exemple de communication

Voici un exemple (incomplet) d'une communication pour ce défi. Dans cet exemple les "<" et ">" indiquent le sens de transfert de chaque message et ne doivent pas être présents dans la communication.

```
< -- Début du défi : Random avec bornes --
< 1011010101011011
< 1011111110101010
> 1011110111001101
< OK
< 10011100101010101
< 10011100101100000
> 10011100101011000
< OK
< 1111100100011
< 11101010101010
> 10001101100001
< OK
```