# Rattrapage EI C#

# Laplace Hugo

## Question 1 :

A . Non le port est défini sur en dur dans le code, le modifier exige donc de modifier le code, le maximum de connexion sur le port est de 1, on ne peut également pas modifier ça.

B. Le serveur accepte la connexion en local (localhost : 127.0.0.1), c’est-à-dire le serveur et le client doivent tourner sur la même machine.

## Question 2 :

A. Le constructeur en privé n’est pas une mauvaise pratique en soi, car le constructeur de la méthode est un DP sémaphore, ce qui évite la double instanciations de la classe.

B. Le fichier log se trouve dans le dossier contenant les fichiers du projet ServeurFindNumber.

Il serait mieux placer dans un dossier contenant tous les logs.

La manière dont le fichier de log est nommé, permet la création d’un fichier par jour, ce qui est plutôt une bonne pratique malgré ça, j’aurais rajouté avant un nom expliquant le fichier, tel que LogServer-date.log.

## Question 3 :

string AdresseIPclient = ((IPEndPoint)clientSocket.RemoteEndPoint).Address.ToString();

int PortIPClient = ((IPEndPoint)this.clientSocket.RemoteEndPoint).Port;

Console.WriteLine("Adresse IP du client : " + AdresseIPclient);

Console.WriteLine("Port du client : " + PortIPClient);

## Question 4 :

Les messages envoyés, sont encodés en ASCII, les lettres avec accents n’étant pas de correspondance ASCII, il faut donc remplacer l’encodage ASCCI, par un autre type d’encodage gérant les accents.

## Question 5 :

Le code est globalement bien fait, il utilise des design pattern pour éviter les doubles instanciations, ce qui est plutôt une bonne pratique.

Une erreur que j’ai remarqué est dans le choix du chemin du fichier log, utilisé la fonction « Path.Combine » plutôt que de le note en dur garanti la création d’un chemin correct quelque soit le systéme d’exploitation, ou alors ajouter un mécanisme de gestion d’erreur.

Le nommage des variables n’est pas trop mal, il faudrait par contre les harmonisés, par exemple que les variables de types string commence toutes par une majuscule…

Également l’utilisations de variables de types var, n’est pas une bonne pratique, car en cas de BUG la variable ne « protège » pas la suite du programme.