

## test\_octets\_corrects

June 20, 2019

Test les octets pour savoir s'ils sont corrects.

On commence à regarder la taille du message reçu avec le bit de parité.

Si c'est 1213 tout va bien.

```
[6]: f = open("citation_recu_CRC.txt", 'r')
message = f.read()
print(len(message))
```

1213

On cherche le nombre d'octet comportant une erreur.

```
[12]: n = 0
for i in range(len(message)):
    temp = bin(ord(message[i]))
    #on normalise à 8 bits
    temp = temp[:2] + '0' * (10-len(temp)) + temp[2-len(temp):]
    bit_parite = temp[2]
    #print(temp,"->",bit_parite)
    nombrede1 = temp[2:].count('1')
    if not ((nombrede1 % 2) and bit_parite):
        n = n + 1
        #print("ERREUR !")
print("Il y a",n,"erreurs dans le message")
print("C'est à dire",int(n / len(message) * 100),"% d'erreurs")
print("Le générateur aléatoire es")
```

Il y a 113 erreurs dans le message

C'est à dire 9 % d'erreurs

```
[ ]:
```

```
[ ]: f.close()
```