TP attaque CPA sur RSA

Objectif du TP : réaliser une attaque de type CPA (Correlation Power Analysis) sur un cible RSA.

Etapes:

1. **Récupérer les données depuis les fichiers**
   Les données fournies sous forme de fichiers textes sont transformées dans la structure appropriée : le modulo est stocké dans un entier, les traces et les messages dans des matrices. On suppose que les valeurs '-1000.0' en fin de chaque fichier de trace sont des marqueurs de fin de mesures, elles sont retirées à la fin.
2. **Calcul de l'exponentiation et de son poids de Hamming**
   hypothèse sur le bit du poids fort de la clé qui est égal à 1(logiquement non nul).
   le i_ème +1 bit de la clé (de valeur 1 ou 0) est calculé à partir de la fonction d'exponentiation (poids de hamming inclu dans le résultat) avec la sous clé déjà calculée à l'étape i.
3. **Calcul des coefficients de corrélation**
   On calcule le coefficient de corrélation pour chaque hypothèse de bit (soit 0 soit 1) à l'aide du module numpy qui utilise la formule du coefficient de corrélation de Pearson.
4. **Validation de l'hypothèse**
   On conserve l'hypothèse qui possède le coefficient de Pearson le plus élevé (donc l'hypothèse la plus probable). Si l'hypothèse vérifiée est 0, alors on passe aux valeurs suivantes des traces (seulement le carré est effectué), sinon on saute une valeur sur les traces (carré et multiplication effectués). On itère ensuite jusqu'aux dernières valeurs de traces pour obtenir la clé finale.

Vérification : comparaison du résultat avec la factorisation.
Résultat :
Nous avons trouvé la clé (voir d_7.txt), qui correspond à la clé trouvée en factorisant N.
On a : d = 0b10111001010110001111010110001