**P 4 : Utilisation de JNI**

**Implémentation**

Après avoir installé les composants nécessaires à l’utilisation du NDK et android studio, j’ai commencé la réalisation du tp.

Le xml se compose de quatre boutons pour afficher let traductions des directions en japonais, de deux autres boutons pour le switch entre read et write en fonction de si l’on cherche à réaliser le calcul sur le nombre en entrée ou un nombre aléatoire, ainsi qu’une TextView pour le display des différents éléments, ou l’entrée du nombre à calculer.

Les event listener sont tous câblés sur la fonction handler « clique » qui détermine l’action à réaliser en fonction de l’id du bouton qui appelle le handler.

Les traductions et calculs sont réalisés par des fonctions natives en c++, qui retourneront toutes une string à décoder et afficher dans la textView display dans le code java.

Dans le cas du bouton read, les tests sur l’input sont réalisés en java, avant d’envoyer l’input parsé en integer aux fonction natives, et les erreurs sont affichées par un text Toast si nécessaire.

**Synthèse**

Ce tp est une bonne introduction à l’utilisation d’android studio, ainsi qu’a l’implémentation de fonctions natives pour réaliser des calculs en c++, qui sont plus rapides que celles réalisés en java. Cela permet de comprendre comment le bytecode d’android Java ou Kotlin interagit avec le code native, et pourquoi il peut être intéressant d’utiliser les eux en même temps.

**Rendus**

