

## Compte rendu TP2

### Méthode d'enrichissement de code graphiques

Encodage de message :

```
samuel@samuel-X71Q: ~/Master2/Compression/QRCode2
samuel@samuel-X71Q:~/Master2/Compression/QRCode2$ g++ enriCode.cpp -o code
samuel@samuel-X71Q:~/Master2/Compression/QRCode2$ ./code
Matrice principale :
0 1 2
1 2 3
Vecteur :
10 9
9 8
8 7

Resultat de la multiplication :
25 22
52 46
samuel@samuel-X71Q:~/Master2/Compression/QRCode2$
```

*Résultat de la multiplication d'une matrice par un vecteur*

## Encodage de Hamming :

```
samuel@samuel-X71Q: ~/Master2/Compression/QRCode2
samuel@samuel-X71Q:~/Master2/Compression/QRCode2$ g++ enriCode.cpp -o c
samuel@samuel-X71Q:~/Master2/Compression/QRCode2$ ./code
Message a encoder :
1
0
1
1

Resultat de l'encodage
0
1
1
0
0
1
1
samuel@samuel-X71Q:~/Master2/Compression/QRCode2$
```

*Encodage d'un message par la méthode de Hamming*

L'encodage de Hamming se fait en réalisant une « multiplication » entre la matrice génératrice et le message à encoder. Ici le message et la matrice génératrice sont des matrices binaires, la multiplication deux éléments (0 ou 1) vaut 0 si les deux éléments valent 0, 1 sinon.

## Code graphique à deux niveaux :



*QRCode avec des modules de 7 pixels (binarisé)*



*QRCode avec message inserer (1011001)*

**Personnalisation d'un code graphique :**



*Image à intégrer en RGB*



*Image en niveau de gris*

**Assemblage de deux images :**



*Image à assembler*



*Le QRCode à assembler*



*Assemblage des deux images*