

## QR-CODE

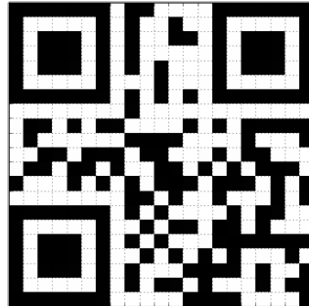


### PRESENTATION

Un QR-code est un codage matriciel d'informations.

**Exemples :**

QR-Code 21 x 21 associé à « HELLO WORLD » :



QR-Code 25 x 25 associé à « CLASSES PREPARATOIRES LA SALLE-HEI » :



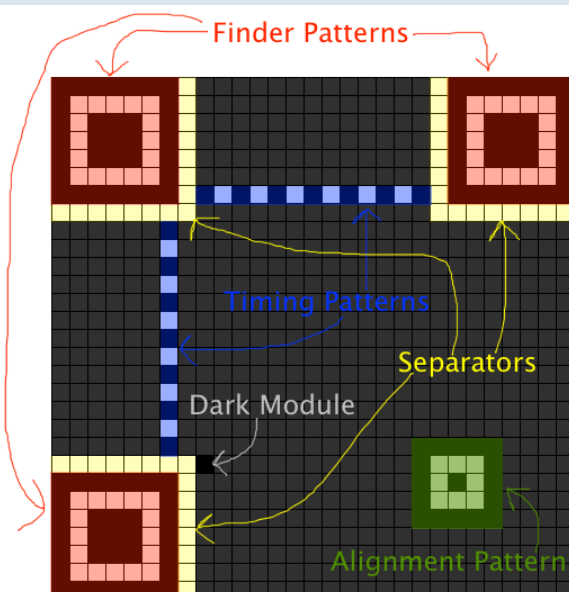
Ces QR-codes peuvent être décodés à l'aide d'une application mobile :



## ELABORATION D'UN QR-CODE

Un QR-code est composé de modules ou motifs (petits carrés) regroupés de la manière suivante :

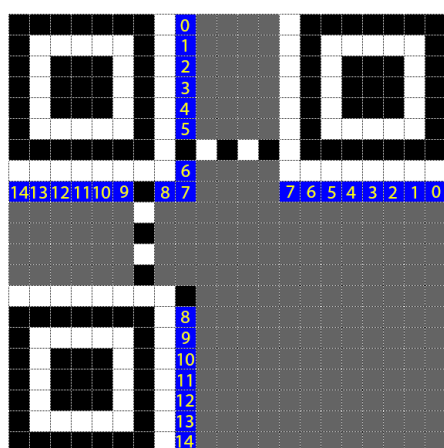
- Des motifs de positionnement : Finder Patterns
- Des séparateurs : Separators
- Des motifs d'orientation : Timing Patterns
- Des motifs d'alignement : Alignment Pattern
- Un module particulier : Dark Module
- Des motifs d'encodage (en gris) contenant :
  - Information à transmettre
  - Redondance d'information (correction d'erreurs)
  - Information de format (masque et redondance)



La création d'un QR-code se fait en 3 étapes :

1. Encoder l'information à transmettre
2. Générer le code correcteur d'erreurs (étant donné un taux de récupération des données)
3. Choisir le « meilleur » masque afin d'optimiser la lecture du QR-code

Notre projet se limite au choix du « meilleur » masque pour un QR-code 21x21 sans motifs d'alignement :



Parmi les motifs d'encodage, l'information à transmettre et la redondance d'information sont en gris alors que l'information de format que l'on confondra avec le choix du masque est en bleu : 15 bits répétés deux fois

## CHOIX DU MEILLEUR MASQUE

Une fois l'information encodée, il ne reste plus qu'à choisir, parmi les huit possibles, le masque qui optimise la lisibilité du QR-code. Des motifs côte à côte de la même couleur rendent en effet plus difficile la lecture.

Pour cela,

- on génère les huit QR-codes possibles
- on attribue, à chacun, un « score » suivant quatre règles de pénalité
- on choisit le masque ayant le « score » le plus faible (le moins pénalisant à la lecture)

## VISUALISATION DES MASQUES

Numéro du masque	Si formule Vraie alors 0 sinon 1	Information de format (bleu)
0	$(Lig + Col) \bmod 2 == 0$	011010101011111
1	$Lig \bmod 2 == 0$	011000001101000
2	$Col \bmod 3 == 0$	011111100110001
3	$(Lig + Col) \bmod 3 == 0$	011101000000110
4	$(Lig // 2 + Col // 3) \bmod 2 == 0$	010010010110100
5	$((Lig * Col) \bmod 2 + (Lig * Col) \bmod 3 == 0$	010000110000011
6	$((Lig * Col) \bmod 2 + (Lig * Col) \bmod 3) \bmod 2 == 0$	010111011011010
7	$((Lig + Col) \bmod 2 + (Lig + Col) \bmod 3) \bmod 2 == 0$	010101111101101

Déclarer une fonction de paramètre un entier de 0 à 7 (numéro du masque) permettant de visualiser (dans la console) le masque sous la forme d'un tableau 21x21 constitué de 0 et de 1.

**Méthode** : déclarer d'abord une fonction retournant une liste de listes de 0 et de 1 pour chaque masque.

**Exemple** :

Pour visualiser la liste de listes `[[1,0],[0,1]]`, on peut utiliser le code suivant qui utilise des chaînes de caractères :

```
8#% Fonction pour visualiser un tableau de 0 et de 1
9def visualiseTab(tab):
10    print('+', '-'*2*len(tab), '+', sep='')
11    for lig in range(len(tab)):
12        print('|', end='')
13        for col in range(0, len(tab)) :
14            if tab[lig][col]==1 :
15                print(chr(9608)*2, end='')
16            else :
17                print(' '*2, end='')
18        print('|', '\n', end='')
19    print('+', '-'*2*len(tab), '+', sep='')
20    return(None)
21#% Test
22tab=[[1,0],[0,1]]
23visualiseTab(tab)
```

## APPLICATION D'UN MASQUE

Un module blanc est codé par 0 et un module noir par 1.

Appliquer un masque (tableau 21x21 de 0 et de 1) revient à effectuer, pour chaque module, un Xor (ou exclusif).

Rappel :

Xor	0	1
0	0	1
1	1	0

Remarque : une nouvelle application du masque aux données modifiées permet de retrouver les données initiales.

donnée	masque	donnéeModifiée = donnée Xor masque	masque	donnéeModifiée Xor masque
0	0	0	0	0
0	1	1	1	0
1	0	1	0	1
1	1	0	1	1

Exemple :

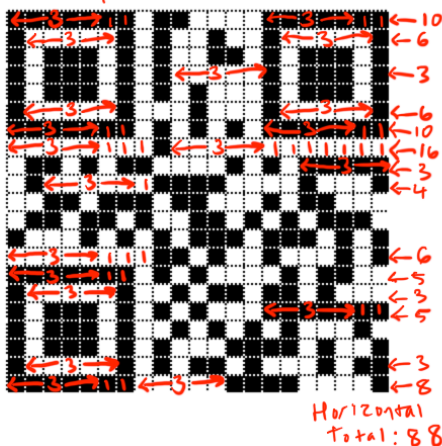
Modules de données	Xor	masque	=	Modules de données modifiées
1 0 1 0 1 1 1 1		0 0 1 0 0 1 0 0		1 0 0 0 1 0 1 1

## REGLES DE PENALITÉ

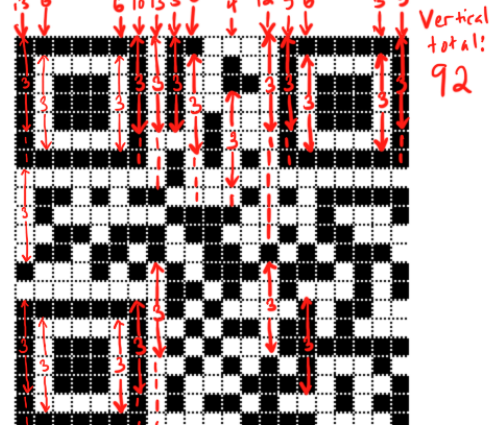
Voici les quatre règles de pénalité :

1. Une pénalité est appliquée pour chaque groupe (horizontal et vertical) d'au moins 5 modules consécutifs de la même couleur. La pénalité est de 3 pour les 5 premiers modules puis de 1 pour chaque module supplémentaire :

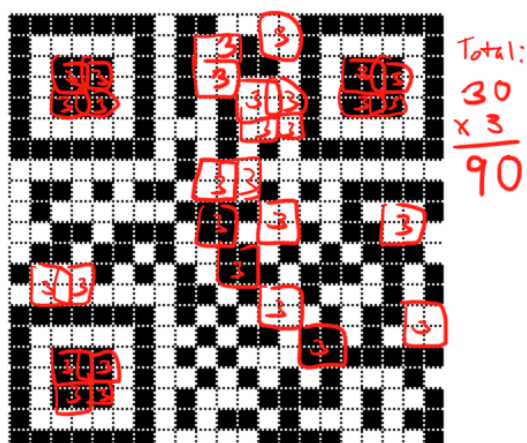
Five consecutive squares get a penalty of 3.  
Each consecutive square after that adds 1 to the penalty.



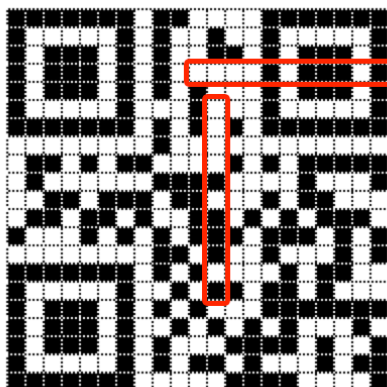
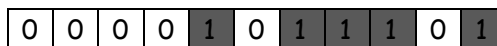
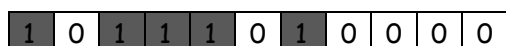
Five consecutive squares get a penalty of 3.  
Each consecutive square after that adds 1 to the penalty.



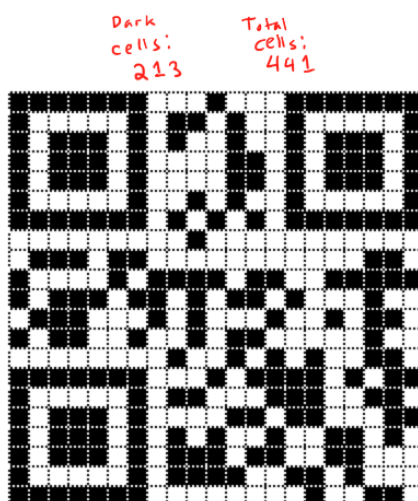
2. Une pénalité de 3 est appliquée pour chaque bloc de 2x2 modules de la même couleur :



3. Une pénalité de 40 est appliquée pour chaque groupe de la forme suivante :



4. Une pénalité est appliquée si la part des modules noirs est trop différente de celle des modules blancs :



Calcul de la pénalité 4 :

- Pourcentage de modules noirs :  $(213 / 441) * 100$  soit environ 48.299 arrondi à 48.3
- Encadrement du pourcentage par des multiples de 5 :  $45 \leq 48.3 \leq 50$
- Cinquième des distances à 50 des bornes de l'encadrement :  $|45 - 50| / 5 = 1$  et  $|50 - 50| / 5 = 0$
- Minimum de ces 2 nombres multiplié par 10 :  $0 * 10 = 0$

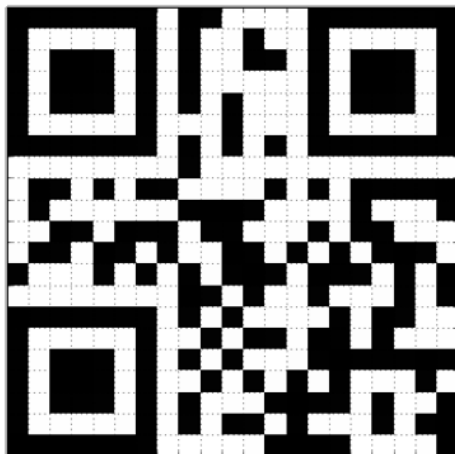
Déclarer une fonction de paramètre une liste de listes (de 0 et de 1) qui retourne la pénalité totale.

**Méthode** : déclarer d'abord une fonction pour chaque pénalité

### EXEMPLE « HELLO WORLD »

Pour ce QR-code, c'est le masque 0 qui a été choisi.

Calculer la pénalité associée au masque 0.

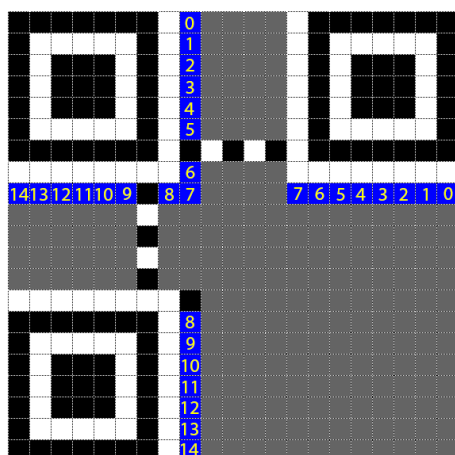


En appliquant le masque 0 (masque utilisé pour ce QR-code), retrouver les données brutes.

**Méthode** : créer un "tableau" 21x21 de 0 et de 1 précisant les modules à modifier lors de l'application du masque.

**Attention, le masque ne s'applique qu'aux modules gris** (information à transmettre et redondance d'information)

L'information de format (en bleu) associée au masque 0 est 01101010101111 ce qui signifie que le module désigné par 14 vaut 0, celui par 13 vaut 1, celui par 12 vaut 1 ... et celui par 0 vaut 1.



Générer les 7 autres QR-code et calculer leur pénalité (pour comprendre le choix du masque 0).

## RECUPERATION DES DONNEES (POUR VOTRE CULTURE)

Les QR-codes peuvent réparer les données manquantes (ou endommagées) grâce au code correcteur d'erreurs.

