

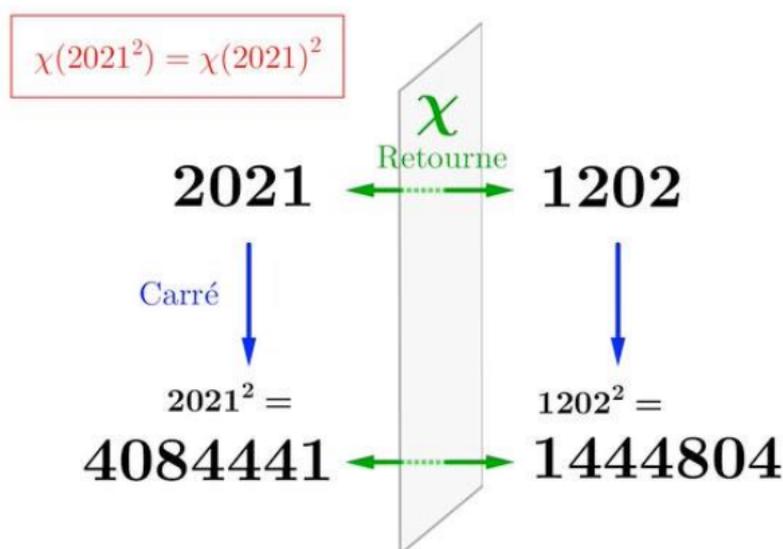
# TP Scratch : une curiosité mathématique de 2021 !

vincent.huvelle@ac-grenoble.fr

Cycle 4

Cette idée de créer ce TP Scratch m'est venue après la lecture d'un tweet de Vincent Pantaloni<sup>1</sup>.

## 1 Une curiosité mathématique



## 2 À la recherche d'autres nombres vérifiant cette curiosité

Tout mathématicien en herbe a immédiatement envie de chercher d'autres nombres vérifiant cette curiosité<sup>2</sup>.

Ce travail peut être fastidieux et décourageant, à première vue, si on te demande de trouver tous les nombres inférieurs à 100 000 vérifiant  $C_1$ .

Pour cela, on va écrire un programme Scratch et laisser travailler l'ordinateur !

### 2.1 Création d'un bloc Reverse

Pour l'étude de cette curiosité, on va exclure naturellement, les nombres palindromes<sup>3</sup>, et par la même occasion les multiples de puissances de 10.

**Objectif du bloc :**

- On entre un nombre en entrée du bloc,
- le script du bloc affecte à une variable `reversenombre` le nombre « retourné ».

#### 2.1.1 Piste noire

Alors au travail ! Réalise l'objectif et rends toi à la prochaine étape.

1. <https://twitter.com/VPantaloni/status/1344976421106704384>  
2. On appellera  $C_1$  cette curiosité  
3. Mot ou groupe de mots qui peut se lire indifféremment de gauche à droite ou de droite à gauche en gardant le même sens (bob, kayak...)

### 2.1.2 Piste rouge

1. Décompose 2021, comme dans l'exemple ci-dessous :

$$721 = 7 \times 100 + 2 \times 10 + 1 \times 1$$

2. À partir de la décomposition de 2021, déduis-en celle de 1202.

3. En programmation, on doit souvent utiliser des variables que l'on pourrait qualifier de « compteurs ». En effet, il va être utile ici, qu'un compteur permette d'obtenir un chiffre du nombre de départ et lui donne la valeur correspondante dans le nombre reverse.

Par exemple pour 2021 :

- Le **1er** chiffre, en partant de la gauche est 2 qui devient  $2 \times 1 = 2 \times 10^{1-1}$  pour le nombre reverse,
- Le **2ème** chiffre est 0 qui devient  $0 \times 10^{2-1}$  pour le nombre reverse,
- Le **3ème** chiffre est 2 qui devient  $2 \times 10^{3-1}$  pour le nombre reverse,
- Le **4ème** chiffre est 1 qui devient  $1 \times 10^{4-1}$  pour le nombre reverse,

4. Quelques blocs utiles pour réussir :



### 2.1.3 Piste bleue

Voici l'algorithme à programmer :

`reversenombre` ← 0

`compteur` ← 0

Répéter **longueur du nombre entré** de fois :

- ajouter le chiffre correspondant au `compteur` du nombre entrée multiplié par la puissance de 10 d'exposant le nombre précédent `compteur`
- ajouter 1 à `compteur`

## 2.2 Test sur les 100 000 premiers nombres

Il suffit maintenant de tester la curiosité sur les 100 000 premiers nombres.

On peut, au fur et à mesure, stocker les nombres vérifiant cette curiosité dans une liste.

### 2.2.1 Piste noire

Au travail!

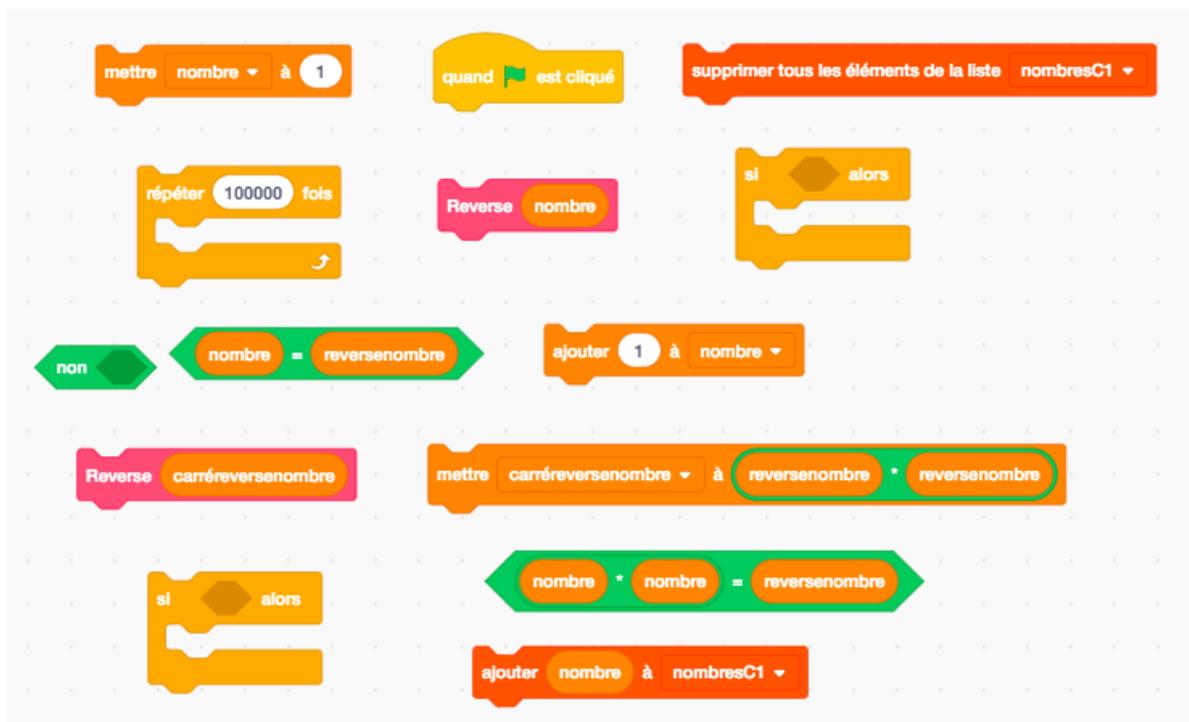
### 2.2.2 Piste rouge

Après avoir créé une liste `nombresC1` et une variable `nombre`, tu peux commencer le script dont les grandes lignes sont :

- Initialisation de la liste et de la variable,
- Faire une boucle qui va permettre de tester si le nombre vérifie C1.

### 2.2.3 Piste bleue

Écris le script en utilisant les blocs ci-dessous.



### 3 Prolongement

Quels sont les seuls chiffres qui composent les nombres vérifiant C1 ? Essaie de justifier...