

# DM1: Encodage d'ensembles

MP2I Lycée Pierre de Fermat

## Consignes

Ce DM est à faire sur papier. Si vous êtes à l'aise en LaTeX, vous pouvez le rédiger sur ordinateur et le rendre par mail. Date de rendu : **vendredi 18 octobre au début du cours.**

Le but de ce problème est d'apprendre à encoder des sous-ensembles finis de  $\mathbb{N}$  par des entiers, et de traduire les opérations ensemblistes en opérations sur les entiers.

**Définition 1.** Soit  $E \subseteq \mathbb{N}$  un ensemble *fini* d'entiers. Le code de  $E$ , noté  $c(E)$ , est défini par :

$$c(E) = \sum_{i \in E} 2^i$$

Par exemple, l'ensemble  $E = \{0, 3, 4\}$  sera encodé par  $c(E) = 2^0 + 2^3 + 2^4 = 1 + 8 + 16 = 25$ .

**Q1.** Pour chaque valeur de  $E$  suivant, donnez  $c(E)$  en décimal **et** en binaire.

- a)  $E = \{1, 2\}$       b)  $E = \{5\}$       c)  $E = \{1, 2, 5\}$       d)  $E = \{0, 1, 2, 3\}$

**Q2.** Soit  $n \in \mathbb{N}$ . Pour chaque valeur de  $E$  suivant, donnez  $c(E)$  en décimal **et** en binaire.

- a)  $E = \{n\}$       b)  $E = \{0, 1, \dots, n-1\}$ .

**Q3.** Donnez les écritures en base 2 des entiers suivants, puis donnez les ensembles qu'ils encodent.

- a) 97      b) 2024

Pour  $n \in \mathbb{N}$ , on note  $d(n)$  l'ensemble encodé par  $n$ . On a donc, pour  $E \subseteq \mathbb{N}$  fini,  $d(c(E)) = E$ .

**Q4.** Donner une expression de  $d(n)$  en fonction de  $n$ .

**Dans toute la suite du sujet**, sauf si explicitement indiqué autrement, on s'intéresse uniquement à des ensembles  $E \subseteq \{0, 1, \dots, 7\}$ , afin de pouvoir représenter les codages des ensembles sur des *entiers non signés sur 8 bits*.<sup>1</sup> On appellera *ensemble codable* tout ensemble  $E \subseteq \{0, 1, \dots, 7\}$ .

Dans le code C de ce problème, le type utilisé pour les codages sera `unsigned char`, qui est le type des entiers non signés sur 8 bits. Ce type est long à écrire, et afin de simplifier la rédaction, on suppose dans toute la suite du sujet que l'on a renommé ce type en `uchar`. En C, on peut utiliser la syntaxe suivante pour effectuer ce renommage :

```
1 typedef unsigned char uchar;
```

**Q5.** Combien d'entiers différents peut-on représenter avec un `uchar`? Lesquels?

<sup>1</sup> Notons que tout ce qui est fait dans cette partie fonctionnerait pour des entiers sur 32 bits, sur 64 bits, etc... mais je vous épargne le fait de devoir écrire des entiers de 64 bits.

