

Cartes perforées



Métier à tisser de Jacquard



Ordinateur Mark 1

**Grace Hopper**

Grace Hopper est une mathématicienne Américaine, née en 1906. Elle grandit en montrant une véritable curiosité pour les sciences et la technologie (par exemple, en démontant les réveils de sa maison les uns après les autres, jusqu'à arriver à en remonter un entièrement... à l’âge de 7 ans). Ses parents encouragent son goût pour les sciences tout en veillant à valoriser ses choix plutôt que suivre la tradition en matière d'éducation des jeunes filles Américaines. Ainsi, Grace sera une des rares femmes de son époque à étudier à l'université de Yale et à obtenir un doctorat de mathématiques.

Pendant la deuxième guerre mondiale, elle entre dans la marine, où elle est nommée lieutenant, et rejoint l'équipe qui travaille sur le **Mark 1**, le premier ordinateur des États-Unis. Grace Hopper fait alors partie du premier groupe capable de programmer cette machine.

Cet ordinateur prend ses instructions et ses données à l'aide de **cartes perforées**, comme cela était prévu sur le métier à tisser de Jacquard environ deux siècles auparavant.

**Comment programmer un ordinateur autrement qu’avec des cartes perforées ?**

Dans les années 1950, Grace Hopper pense que l'informatique devrait être accessible à un plus grand nombre de personnes, et pas seulement aux spécialistes des ordinateurs et aux mathématiciens. Elle défend l'idée que les cartes perforées sont un frein au développement de l’informatique et travaille sur un projet de « **langage de programmation** » qui serait proche de l'anglais, et qui faciliterait l'écriture des programmes.

La difficulté est alors de concevoir le programme capable de transformer des instructions données dans un langage proche de l'anglais, facilement compréhensible par un **humain**, en instructions compréhensibles par la **machine**…

**Le langage machine**

Le **langage machine**, ou **code machine**, est la suite de [**bits**](https://fr.wikipedia.org/wiki/Bit_%28informatique%29) **(succession de 1 et de 0)** qui est interprétée par le [**processeur**](https://fr.wikipedia.org/wiki/Processeur) d'un [**ordinateur**](https://fr.wikipedia.org/wiki/Ordinateur) exécutant un [**programme informatique**](https://fr.wikipedia.org/wiki/Programme_informatique).

Ce programme est composé d'[**instructions**](https://fr.wikipedia.org/wiki/Instruction_informatique) et de **données** à traiter. Ces instructions et ces données sont codées avec des 0 et des 1(langage binaire, seul langage qu’une machine puisse comprendre).

Plus on juxtapose de bits, plus on peut représenter d’éléments.

Avec un seul bit (2x1), on peut coder **2 éléments** (blanc et noir par exemple) : code « 0 » pour le blanc, code « 1 » pour le noir

Avec 2 bits (2x2), on peut coder **4 éléments,** en combinant les « 1 » et les « 0 », avec les codes 00 ; 01 ; 10 ; 11. Par exemple, « 00 » pour coder le signe +, « 01 » pour coder le signe -, 10 pour coder le signe x, 11 pour coder le signe ÷.

Avec 3 bits (2x2x2), on peut coder **8 éléments,** en combinant les « 1 » et les « 0 », avec les codes 000 ; 001 ; 010 ; 011 ; 100 ; 101 ; 110 ; 111.

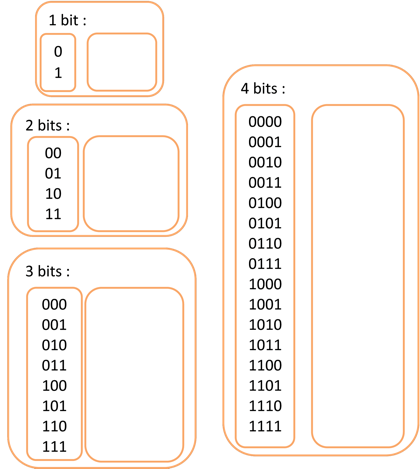
Avec 4 bits (2x2x2x2), on peut coder 16 éléments, en combinant les 1 et les 0 (0000 ; 0001 ; …), et ainsi de suite.

Ce codage binaire permet de représenter toute sorte de données.

**Naissance du « bug » informatique**

Grace Hopper n'est pas célèbre uniquement pour avoir rendu les ordinateurs plus accessibles aux non-spécialistes. Un jour, alors qu'elle programme ses ordinateurs avec des cartes perforées, l'un d'eux tombe en panne. Allant voir ce qui a pu se produire, elle trouve une mite coincée dans un des trous d'une carte perforée. Elle colle alors sur son rapport le morceau de carte avec un commentaire désignant l'insecte comme coupable de la panne. « Insecte » se disant « bug » en Anglais, ce terme a  été utilisé auparavant par les scientifiques, comme Thomas Edison, pour désigner une erreur due à l'intervention non désirée d'une de ces petites bêtes dans leurs expériences. En l'appliquant à l'exécution échouée d'un programme informatique, Grace en a fait le terme le plus utilisé pour qualifier cette situation.

Extrait du carnet de notes de Grace Hopper où elle a collé « the bug ».



Noir

Blanc

**Exercice 3 « 3bits » :** Ecris un exemple de liste dont tous les éléments peuvent être codés sur 3 bits.

Code l’algorithme suivant en langage binaire.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| … … | … … | … … | … … | … … | … … | … … |

**Exercice2 « 2bits »**: Avec 2 bits, on peut coder, par exemple, les 4 figures géométriques de base : Carré, Cercle, Rectangle, Triangle.

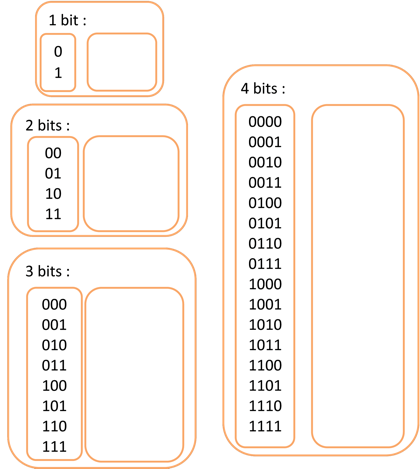
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

**Exercice1 « 1bit »**: Si on code le noir par la valeur « 0 », et le blanc par la valeur « 1 », en coloriant la grille on obtient………………….

**Exercice 5, défi :** détermine combien de bits il faut juxtaposer au minimum pour pouvoir coder chacune des 26 lettres de l’alphabet.

Tu peux faire tes essais de combinaisons ci-dessous.

...



0

1

2

3

4

5

6

7  
8  
9

+

-

x

/

=

C

**Exercice 4:** Avec 1 bit, un peut coder une liste de deux éléments (noir et blanc par exemple dans l’exercice 1). Avec 2 bits, on peut coder une liste de 4 éléments, avec 3 bits une liste de 8 éléments. Combien faut-il de bits pour coder les éléments de la liste suivante : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, x, /, =, C.

Code chacun des ces éléments en langage binaire (avec des 1 et des 0).

A l’aide de la table de correspondance ci-dessus, encode le message ci-dessous en langage binaire.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Message en clair | D | I | X |  | M | I | N | U | T | E | S |
| Message codé en binaire |  |  |  |  |  |  |  |  |  |  |  |

**Exercice 6 :** Table de correspondance pour coder chacune des 26 lettres de l’alphabet.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 bits | 10000 | 10001 | 10010 | 10011 | 10100 | 10101 | 10110 | 10111 |
| Caractère | Q | R | S | T | U | V | W | X |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 bits | 00000 | 00001 | 00010 | 00011 | 00100 | 00101 | 00110 | 00111 |
| Caractère | A | B | C | D | E | F | G | H |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 bits | 01000 | 01001 | 01010 | 01011 | 01100 | 01101 | 01110 | 01111 |
| Caractère | I | J | K | L | M | N | O | P |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 bits | 11000 | 11001 | 11010 | 11011 | 11100 | 11101 | 11110 | 11111 |
| Caractère | Y | Z | . | (espace) | aucune signification (on peut, si on le souhaite, s’en servir pour encoder d’autres signes de ponctuation) | | | |