

Codage et Programmation

Cycle 3

A word cloud centered around the word "Informatique" (Computer Science). The words are in various colors and sizes, representing related concepts and terms. The most prominent word is "Informatique" in large orange letters. Other significant words include "Langage" (Language) in blue, "Algorithme" (Algorithm) in orange, "Ordinateur" (Computer) in pink, and "Difficultés" (Difficulties) in green. Smaller words include "Binaire" (Binary), "Ludique" (Playful), "Notation", "Logique" (Logic), "définir" (define), "Complicqué" (Complicated), "Programme", "Galère" (Struggle), "Enigme" (Mystery), "Précision", "Chinois", "Organisation", "Chiffres" (Numbers), "Logiciel" (Software), "Ennui" (Boredom), "Déplacements" (Movements), "Matériel" (Hardware), "Code", and "Scratch".

Binaire
Ludique
Notation
Logique
Langage
définir
Complicqué
Programme
Galère
Enigme
Précision
Algorithme
Informatique
Ordinateur
Chinois
Organisation
Chiffres
Difficultés
Ennui
Déplacements
Matériel
Code
Scratch

1. Que disent les programmes 2015 ?

- **Mathématiques**

En complément de l'usage du papier, du crayon et de la manipulation d'objets concrets, **les outils numériques sont progressivement introduits**. Ainsi, l'usage de logiciels de calcul et de numération permet d'approfondir les connaissances des propriétés des nombres et des opérations comme d'accroître la maîtrise de certaines techniques de calculs.

De même, **des activités géométriques peuvent être l'occasion d'amener les élèves à utiliser différents supports de travail : papier et crayon, mais aussi logiciels de géométrie dynamique, d'initiation à la programmation ou logiciels de visualisation de cartes, de plans.**

Que disent les programmes 2015 ?

- **Nombres et Calculs**

De même, si la maîtrise des techniques opératoires écrites permet à l'élève d'obtenir un résultat de calcul, la construction de ces techniques est l'occasion de retravailler les propriétés de la numération et de **rencontrer des exemples d'algorithmes complexes.**

Calcul posé

- Mettre en oeuvre un algorithme de calcul posé pour l'addition, la soustraction, la multiplication, la division.

Que disent les programmes 2015 ?

- Espace et Géométrie

Les activités spatiales et géométriques sont à mettre en lien avec les deux autres thèmes : résoudre dans un autre cadre des problèmes relevant de la proportionnalité ; utiliser en situation les grandeurs (géométriques) et leur mesure.

Par ailleurs, elles constituent des moments privilégiés pour une **première initiation à la programmation notamment à travers la programmation de déplacements ou de construction de figures.**

Que disent les programmes 2015 ?

- **(Se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations**
 - Se repérer, décrire ou exécuter des déplacements, sur un plan ou sur une carte.
 - Accomplir, décrire, **coder des déplacements dans des espaces familiers.**
 - Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran.
 - Vocabulaire permettant de définir des positions et des déplacements.
 - Divers modes de représentation de l'espace.

Que disent les programmes 2015 ?

- Effectuer des tracés correspondant à des relations de perpendicularité ou de parallélisme de droites et de segments.
- Déterminer le plus court chemin entre deux points (en lien avec la notion d'alignement).
- Déterminer le plus court chemin entre un point et une droite ou entre deux droites parallèles (en
- lien avec la perpendicularité).

Exemples de matériels : papier/crayon, **logiciels** de géométrie dynamique, **d'initiation à la programmation**, logiciels de visualisation de cartes, de plans.

Que disent les programmes 2015 ?

- Sciences et technologie

Matière, mouvement, énergie, information

Identifier un signal et une information

Introduire de façon simple la notion de signal et d'information en utilisant des situations de la vie courante : feux de circulation, voyant de charge d'un appareil, alarme sonore, téléphone...

Élément minimum d'information (oui/non) et représentation par 0, 1.

Que disent les programmes 2015 ?

Les élèves découvrent l'algorithme en utilisant des logiciels d'applications visuelles et ludiques.

- Ils exploitent les moyens informatiques en pratiquant le travail collaboratif. Les élèves maîtrisent le fonctionnement de logiciels usuels et s'approprient leur fonctionnement.

Dans ces classes, l'investigation, l'expérimentation, l'observation du fonctionnement, la recherche de résolution de problème sont à pratiquer afin de solliciter l'analyse, la recherche, et la créativité des élèves pour répondre à un problème posé.

Leur solution doit aboutir la plupart du temps à une réalisation concrète favorisant la manipulation sur des matériels et l'activité pratique.

L'usage des outils numériques est recommandé pour favoriser la communication et la représentation des objets techniques.

2. Algorithmes, codes, programmes ?

```
base64.cc
31 void base64_encode(const uint8_t * data, size_t length, char * dst)
32 {
33     size_t src_idx = 0;
34     size_t dst_idx = 0;
35     for (; (src_idx + 2) < length; src_idx += 3, dst_idx += 4)
36     {
37         uint8_t s0 = data[src_idx];
38         uint8_t s1 = data[src_idx + 1];
39         uint8_t s2 = data[src_idx + 2];
40
41         dst[dst_idx + 0] = charset[(s0 & 0xfc) >> 2];
42         dst[dst_idx + 1] = charset[((s0 & 0x03) << 4) | ((s1 & 0xf0) >> 4)];
43         dst[dst_idx + 2] = charset[((s1 & 0x0f) << 2) | (s2 & 0xc0) >> 6];
44         dst[dst_idx + 3] = charset[(s2 & 0x3f)];
45     }
46
47     if (src_idx < length)
48     {
49         uint8_t s0 = data[src_idx];
50         uint8_t s1 = (src_idx + 1 < length) ? data[src_idx + 1] : 0;
51
52         dst[dst_idx++] = charset[(s0 & 0xfc) >> 2];
53         dst[dst_idx++] = charset[((s0 & 0x03) << 4) | ((s1 & 0xf0) >> 4)];
54         if (src_idx + 1 < length)
55             dst[dst_idx++] = charset[((s1 & 0x0f) << 2)];
56     }

```

Line 31, Column 55

Spaces: 4

C++

Histoire de la programmation

- Pour mieux comprendre ce qu'est un algorithme et sa formalisation en programme, il est intéressant avec les élèves de plonger dans l'Histoire.
- Voici quelques grandes étapes et quelques personnages importants à placer sur vos frises chronologiques.

2.a Histoire de la programmation

- Chacun de ces personnages, d'Al Khwarizmi (né vers 780) à Grace Hopper (1906-1992), alors qu'ils vivent à des périodes où les ordinateurs n'ont pas encore envahi le monde, contribuent à leur manière à l'élaboration des 4 concepts fondateurs de l'informatique: **algorithme, langage, machine, information.**

AL KHWARIZMI (783 -850)

<http://www.universcience.tv/video-l-histoire-d-al-khwarizmi-ou-presque-14906.html>

Charles BABBAGE (1791-1871)

La machine à interpréter n'importe quel algorithme

<https://pixees.fr/lhistoire-de-charles-babbage-ou-presque/>

Ada LOVELACE (1815-1852) la première programmeuse

<http://www.universcience.tv/video-l-histoire-d-ada-lovelace-ou-presque-12314.html>

Alan TURING (1912-1954)

<http://www.universcience.tv/video-001-contre-enigma-5507.html>

Grace HOPPER (1906-1992)

<http://www.universcience.tv/video-l-histoire-de-grace-hopper-ou-presque-12230.html>

2.b les notions

Le concept d'Algorithme

Définition: c'est une « recette » qui permet de résoudre un problème ou de faire quelque chose de manière systématique.

Un algorithme comporte un début et une fin.

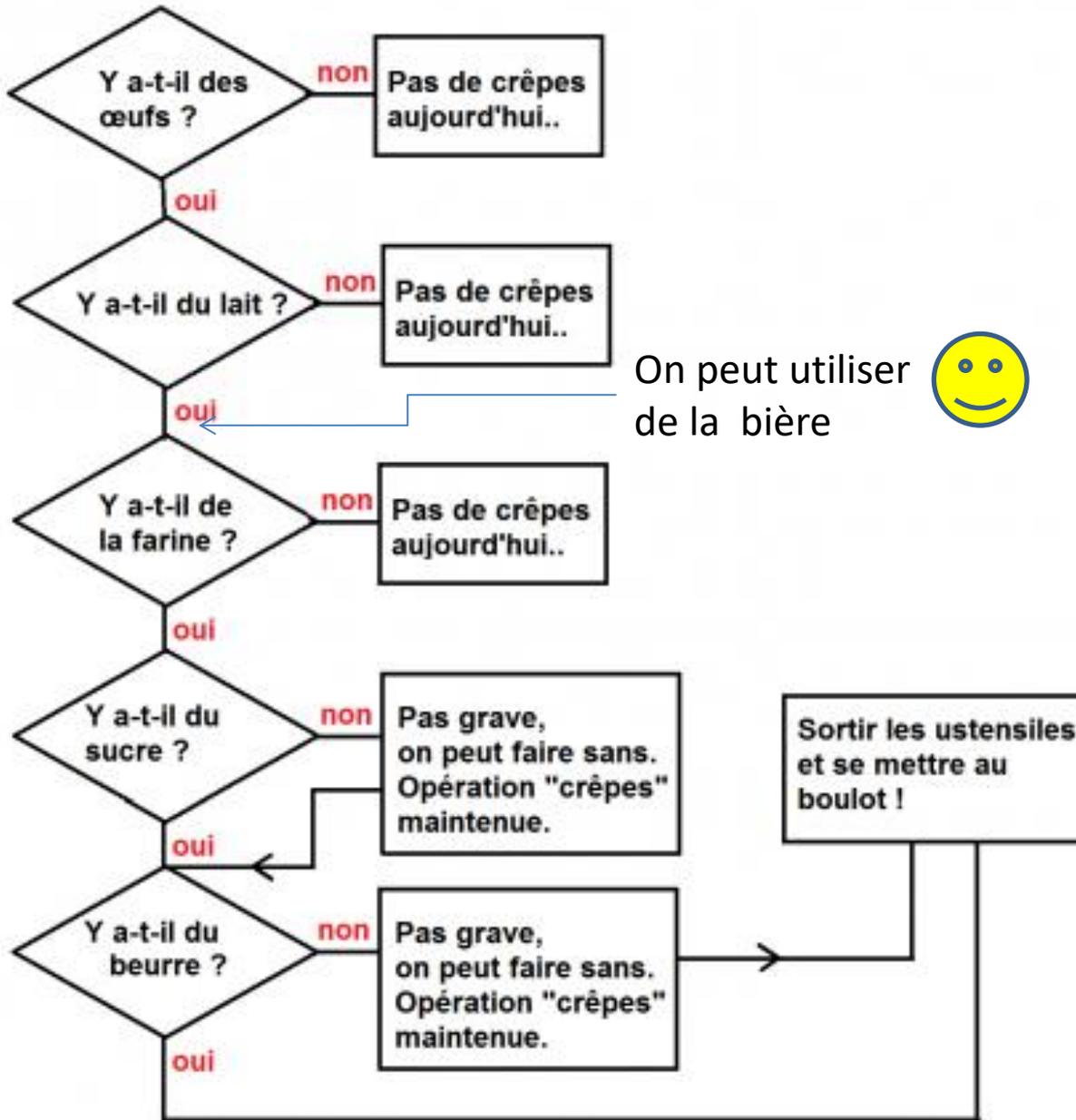
Pendant l'exécution de l'algorithme, des données (**informations**) sont stockées.

Certaines données peuvent changer pendant la réalisation de l'algorithme : ce sont les **variables**.

L'algorithme est composé d'étapes appelées **instructions**.

Quelques exemples d'algorithmes

- Un exemple de problème qui nous concerne tous est celui de la cuisine : vous êtes dans une cuisine, vous trouvez du riz, comment le cuire ?
- **Voici une marche à suivre simple :**
- remplir une casserole d'eau ;
- y ajouter une pincée de sel ;
- la mettre sur le feu ;
- attendre l'ébullition de l'eau ;
- mettre le riz dans la casserole ;
- le laisser cuire 10 à 15 minutes ;
- égoutter le riz.



On peut utiliser de la bière



Introduction des conditions
Si...alors....sinon.....

Le concept de programme

- Le **programme** est la version « écrite » de l'algorithme.
- Un programme informatique est une succession d'instructions exécutable par l'ordinateur.
- Pour réaliser un algorithme, on peut utiliser des programmes différents.

- Pour qu'un programme soit lisible par tout le monde, il faut donc un langage commun.
- Ce langage permettra de traduire les instructions du programme à la machine pour qu'elle puisse agir sur les objets (une photo par exemple) ou sur des données mathématiques qui sont présentes dans l'algorithme.

Mais alors, le code c'est quoi?

Vidéo un jour, une question
C'est quoi le code informatique?

<https://www.youtube.com/watch?v=2UqIJVQE7o>

Définitions

- Coder à plusieurs significations et ambigüités :

Le verbe Coder est souvent assimilé à la modification d'un message pour le rendre illisible aux autres et à le décoder pour le retrouver, on appelle ça le **chiffrement**.

Le **codage**, c'est la programmation informatique, ce qui permet l'écriture des programmes pour développer des logiciels, ou une page web, par exemple, mais aussi pour dire à une machine ce **qu'elle doit faire**, la guider. ... Le vrai mot français est « **programmer** ».

On peut aussi utiliser le verbe coder pour **représenter une information à l'aide de symboles** (par exemple écrire un texte en binaire à l'aide de 0 et de 1)

Le codage Binaire : 011001

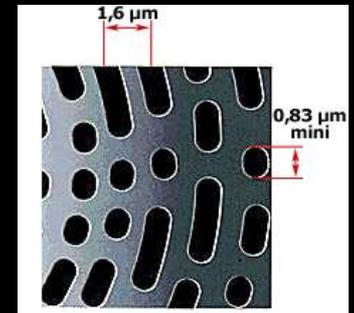
- Vers la fin des années 30, Claude Shannon démontra qu'à l'aide de « contacteurs » (interrupteurs) fermés pour « vrai » et ouverts pour « faux » il était possible d'effectuer des opérations logiques en associant le nombre 1 pour « vrai » et 0 pour « faux ».
- Ce codage de l'information est nommé **base binaire**. C'est avec ce codage que fonctionnent les ordinateurs. Il consiste à utiliser deux états (représentés par les chiffres 0 et 1) pour coder les informations.

Le Bit

Le terme **bit** (*b* avec une minuscule dans les notations) signifie « **binary digit** », c'est-à-dire 0 ou 1 en numérotation binaire. Il s'agit de la plus petite unité d'information manipulable par une machine numérique.

Il est possible de représenter physiquement cette information binaire :

- par un signal électrique, magnétique ou lumineux , qui, au-delà d'un certain seuil, correspond à la valeur 1 ;
- par des aspérités géométriques dans une surface ;



Le Bit

Avec un bit il est ainsi possible d'obtenir deux états : soit 1, soit 0. Grâce à 2 bits, il est possible d'obtenir quatre états différents ($2*2$) :

0	0
0	1
1	0
1	1

Avec 3 bits, il est possible d'obtenir huit états différents ($2*2*2$) :

Val	Vaale
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Dans un nombre binaire, la valeur d'un bit, appelée **poids**, dépend de la position du bit en partant de la droite.

A la manière des dizaines, des centaines et des milliers pour un nombre décimal, le poids d'un bit croît d'une puissance de deux en allant de la droite vers la gauche comme le montre le tableau suivant :

Nombre binaire	1	1	1	1	1	1	1	1
Poids	$2^7 = 128$	$2^6 = 64$	$2^5 = 32$	$2^4 = 16$	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$

L'OCTET

L'**octet** (en anglais *byte* ou *B* avec une majuscule dans les notations) est une unité d'information composée de 8 bits. Il permet par exemple de stocker un caractère, tel qu'une lettre ou un chiffre.

Ce regroupement de nombres par série de 8 permet une lisibilité plus grande, au même titre que l'on apprécie, en base décimale, de regrouper les nombres par trois pour pouvoir distinguer les milliers. Le nombre « 1 256 245 » est par exemple plus lisible que « 1256245 ».

Pour un octet, le plus petit nombre est 0 (représenté par huit zéros 00000000), et le plus grand est 255 (représenté par huit chiffres « un » 11111111), ce qui représente 256 possibilités de valeurs différentes.

- KiloOctets, MégaOctets

- Beaucoup d'informaticiens ont appris que 1 kilooctet valait 1024 octets. Or, depuis décembre 1998, l'organisme international *IEC* a statué sur la question
Voici donc les unités standardisées :
- Un kilooctet (ko ou kB) = 1000 octets
- Un Mégaoctet (Mo ou MB) = 1000 ko = 1 000 000 octets
- Un Gigaoctet (Go ou GB) = 1000 Mo = 1 000 000 000 octets
- Un Téraoctet (To) = 1000 Go = 1 000 000 000 000 octets



- Pour représenter un texte
- Chaque caractère d'un alphabet (minuscule, majuscule, chiffre, autre caractère)est représenté par une suite de 8 bits .
- Exemple : le mot « LE » est représenté de la façon suivante :
- 01001100 de la lettre « L » (76) et suivi de 01100101 pour le « E » (69) soit 0100110001100101
- On appelle ce système le système UNICODE dérivé du système ASCII.

- Pour représenter une image :
- La représentation d'une image est plus difficile car elle repose sur un découpage de l'image en pixels puis par un codage de la teinte de chaque pixel. Donc une image en noir et blanc est moins lourde qu'une image en couleur puisque le noir =1 et le blanc = 0.
- En niveaux de gris, chaque pixel est représenté selon sa teinte en 00000001 si gris foncé, et 11111110 si gris clair et 11111111 si blanc
- Pour les images en couleur chaque pixel est représenté par 3 nombres correspondant à la quantité de rouge, de bleu et de vert le RGB.

	1	2	3	4	5	6	7	8	9	10
1										
2				■	■	■	■			
3			■					■		
4		■							■	
5			■				■			
6										
7			■				■			
8		■			■	■				■
9			■					■		
10				■	■	■	■			
11										

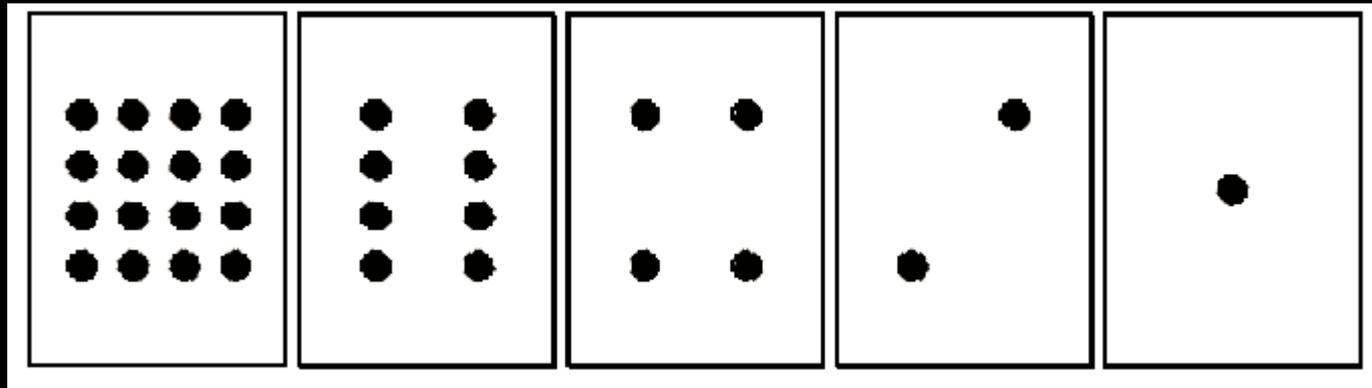
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1	0	0
0	1	0	0	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1	0	0
0	1	0	0	1	1	0	0	1	0	0
0	0	1	0	0	0	0	1	0	0	0
0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

Quelques activités à réaliser avec les élèves

- Les activités « débranchées »

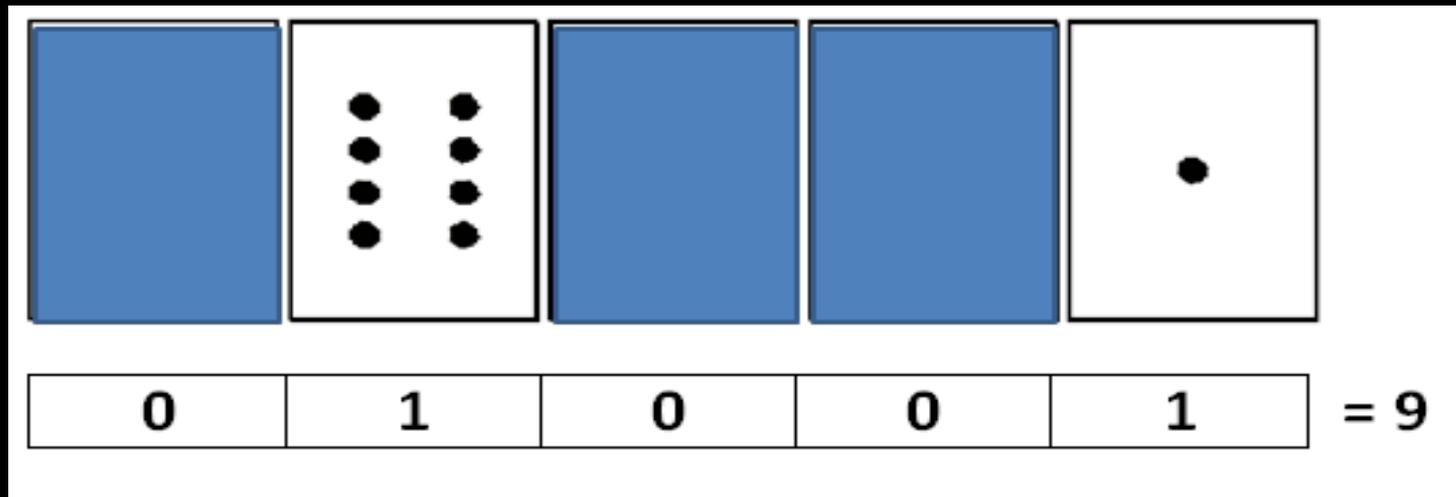
L'écriture binaire des nombres

- Activités 1 : apprendre à compter
- Découvrir le système binaire 0 et 1 qu'utilisent les ordinateurs
- Matériel : 5 cartes



- Obtenir les nombres suivants en retournant des cartes : 10, 3, 12, 31

- **Activité 2 : Travailler en binaire**
- Utiliser le système binaire pour représenter des nombres



- Combien font 10101? Et 11111?
- Trouver la représentation du jour de son anniversaire.
- On peut par la suite imaginer d'autres codes avec des symboles différents.

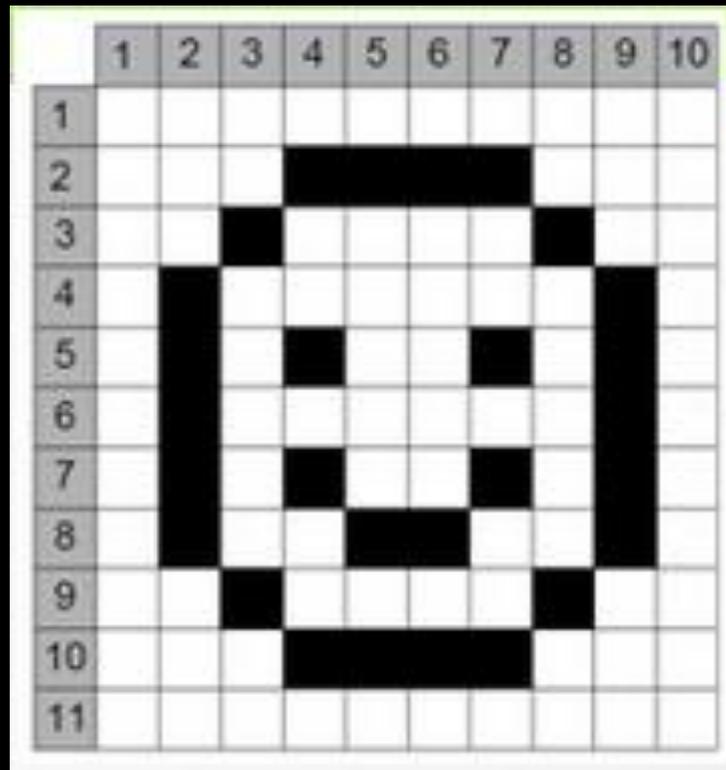
- **Activité 3 : envoyer un message**

Ce petit garçon utilise les lumières du sapin de Noël pour attirer l'attention de la personne située dans l'immeuble d'en face. Il est coincé dans sa chambre et ne peut en sortir.

En utilisant un code simple reprenant le système binaire, trouve le message.

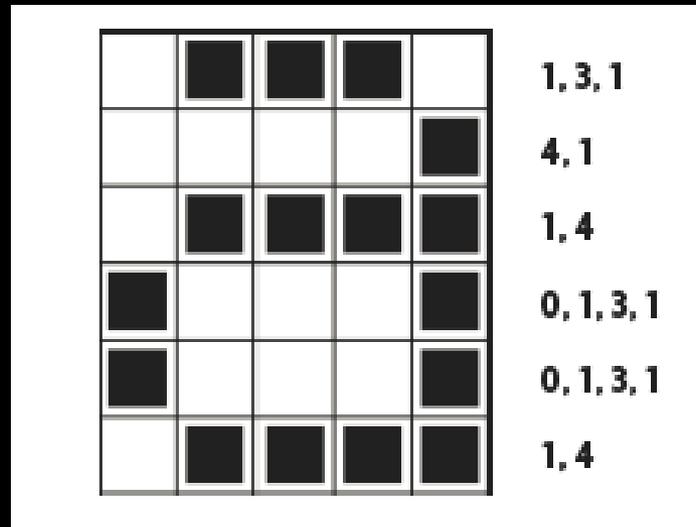
1	2	3	4	5	6	7	8	9	10	11	12	13
a	b	c	d	e	f	g	h	i	j	k	l	m
14	15	16	17	18	19	20	21	22	23	24	25	26
n	o	p	q	r	s	t	u	v	w	x	y	z

- **Activité 4 : représenter une image en binaire**
- On peut assimiler cette activité à du pixel art.
- On choisit bien sûr des images en noir et blanc.



- Représenter des images plus grandes représenterait une somme considérable de calculs pour l'ordinateur. Alors il va compresser ces images en codant différemment.

- Exemple :



L'image ci-dessus nous montre comment une image peut être représentée par des nombres.

La première ligne contient un pixel blanc, trois noirs puis un blanc. Ainsi, la première ligne est représentée par 1, 3, 1.

Le premier nombre représente toujours le nombre de pixels blancs. Si le premier pixel est noir, la ligne commencera par un **0**.

Les algorithmes

- Le jeu de nim

Vidéo le jeu de nim . ROB'O D'EVIAN

https://www.youtube.com/watch?v=M_dX8l_oMH0

- Le jeu de Nim utilise des algorithmes qui mettent en oeuvre la stratégie gagnante.

L'utilisation du « **si...alors** » est essentielle dans le raisonnement scientifique. Le choix de faire vivre avec son corps cette démarche permet d'appréhender un raisonnement abstrait. C'est pourquoi, l'approche pédagogique proposée permet aux élèves de découvrir l'algorithme en utilisant des applications visuelles et ludiques.

L'intention pédagogique est d'aborder la notion d'algorithme à partir d'une information qui permet d'organiser une série de données ou d'objets selon une relation d'organisation déterminée. L'approche de l'algorithme de tri procède par comparaisons successives. « **Si...alors** »

Les déplacements

- Le jeu du robot idiot
- Un enfant joue le robot et les autres enfants vont devoir lui donner des instructions de manière à ce qu'il réalise certaines actions. Attention, un robot ne fait que ce qu'on lui dit de faire... et rien d'autres!

Objectifs :

- Comprendre et expérimenter la notion d'algorithme
- Comprendre qu'un ordinateur ne fait que ce qu'on lui dit de faire et rien d'autres
- Découper une action complexe en actions simples et non ambiguës

Vidéo le jeu du robot idiot

<https://pixees.fr/dis-maman-ou-papa-cest-quoi-un-algorithme-dans-ce-monde-numerique-%E2%80%A8/>

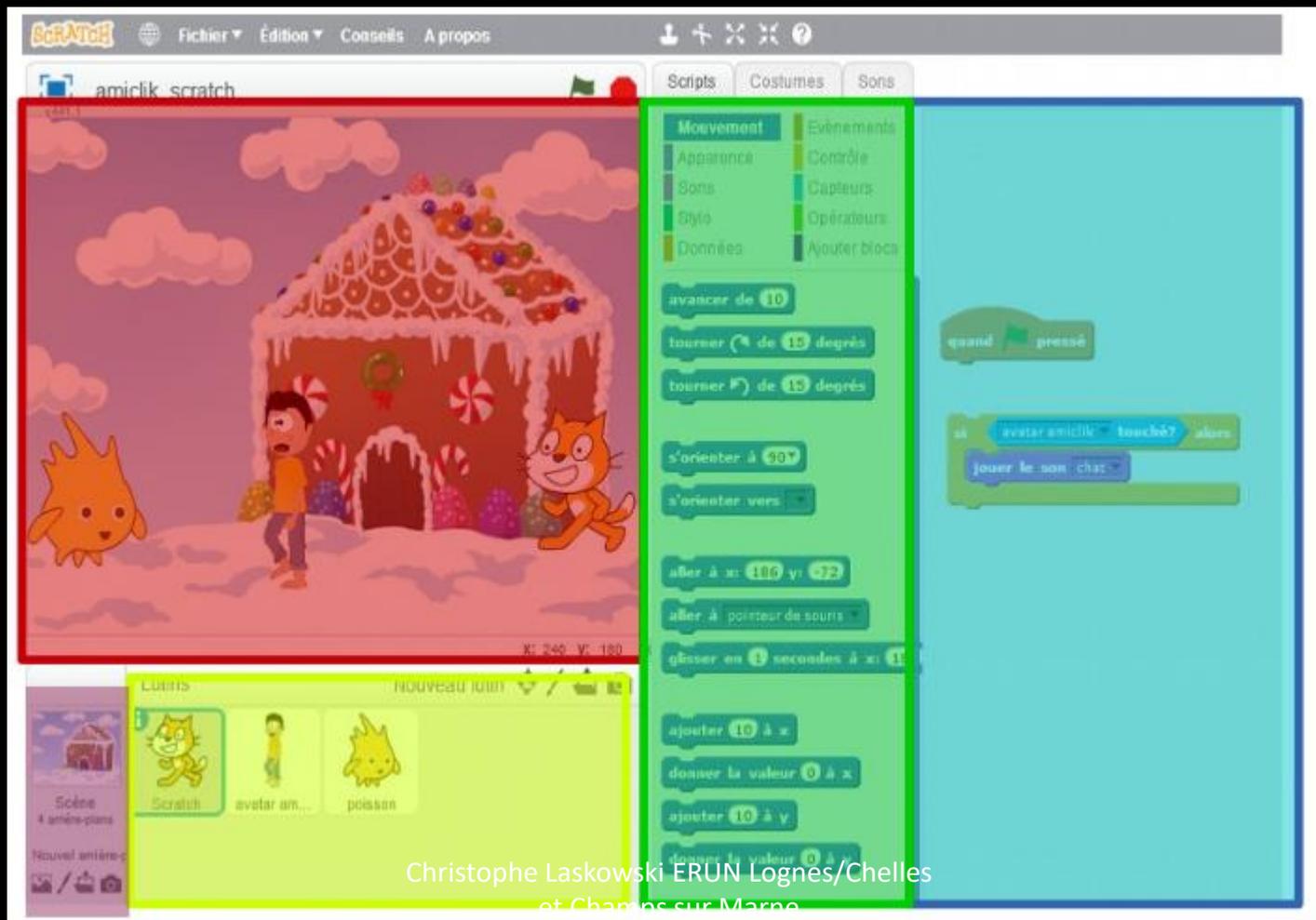
Les activités « branchées »

L'utilisation de logiciels

- Utilisation de logiciels type Scratch, Tuxbot, Run Marco, studio code.org



- Le rendu visuel
- Les blocs
- Les sprites
- Les scripts
- Les fonds d'écrans



Toutes les actions de position du lutin:
Avancer, tourner, déplacer, rebondir sur le bord,

Quand... Touche appuyée, lutin cliqué, envoi message à toutes les actions.

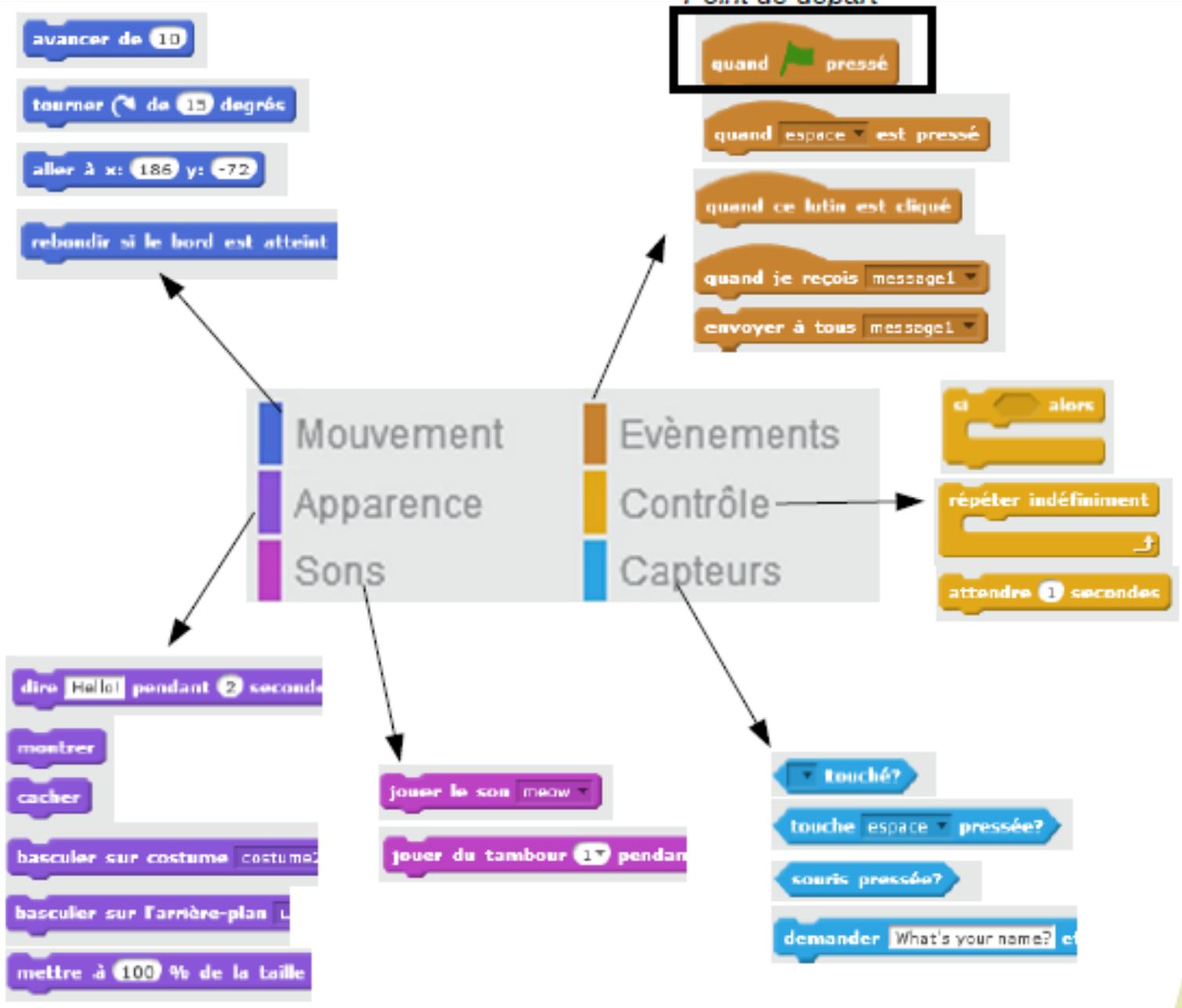
Affichage : costume du lutin, fond d'écran, taille sprite, dire un mot...

Jouer le son : son, instrument, volume

Eléments intégrés dans conditions : touche, couleur, souris



Conditions :
Si...alors, répéter événement, attendre.



Les robots



Sitographie

- <https://code.org/> Code .org
- <https://pixees.fr/> Pixees et Class'Code
- <http://eduscol.education.fr/primabord/codage>
- <http://www.fondation-lamap.org/fr/page/34536/1-2-3-codez-espace-enseignants> La main à la pâte
- <https://scratch.mit.edu/> Scratch