

La programmation expliquée aux enfants

Fiche outil

Mots clés : # Fiche outils # Programmation

Âge : enfants

Public : Enfants

Niveau : Débutant

Objectifs pédagogiques

Découvrir ce qu'est la programmation, ses principes de bases et être capable de les décrire simplement.

Déroulé

- I. Est-ce qu'un ordinateur est intelligent ?
- II. Vous avez dit programmation ?
- III. Qu'est ce que le langage informatique ?
- IV. L'algorithme

I. Est-ce qu'un ordinateur est intelligent ?

[Poser la question : Qui croit qu'il est plus intelligent qu'un ordinateur et faire un vote à main levée en demandant à chacun d'expliquer sa position.]

Puis demander à un des participants d'exécuter une série d'instructions :

- *S'asseoir, se lever plusieurs fois d'affilée*
- *Avancer d'un pas, puis d'un autre pas, et ainsi de suite jusqu'à arriver devant le mur et continuer de demander au participant d'avancer d'un pas*

Puis demander aux autres enfants, si exécuter ces tâches requièrent de l'intelligence.

Explication :

Un ordinateur calcule très vite, il peut répéter une action plusieurs millions de fois sans se lasser, mais il n'est pas intelligent.

La différence entre un humain et un ordinateur est que face à une situation nouvelle, l'être humain peut s'adapter : il essaiera de trouver des similitudes entre son expérience et cette nouvelle situation, expérimentera, fera des suppositions, bref il peut improviser.

Alors qu'un ordinateur lui est incapable d'agir en dehors de ce pourquoi il a été programmé. C'est d'ailleurs un des enjeux du développement des intelligences artificielles.

II. Vous avez dit programmation ?

Un ordinateur, on vient de le voir, ne fait qu'exécuter les instructions qu'on lui a données.

Derrière chaque programme de l'ordinateur, une personne lui a dit quoi faire et comment le faire. Et cette personne qui lui a dit quoi faire, c'est un programmeur.

Que ça soit pour créer un jeu vidéo, ou un site internet ou même une application de téléphone : il y a toujours un ou plusieurs développeurs qui ont expliqué aux ordinateurs, téléphones, tablettes quoi faire.

III. Qu'est-ce que le langage informatique ?

[Poser la question : qui sait quelle langue parle l'ordinateur ?]

Alors le souci quand on veut expliquer quoi faire à un ordinateur, c'est qu'il parle une langue qui s'appelle le binaire : des 0 et des 1 et rien d'autre !

Le binaire, l'ordinateur le comprend très bien mais pour les humains, c'est compliqué à parler.

[Voici un petit exemple :

J'ai un ami espagnol qui parle espagnol mais qui ne parle pas français et moi je parle français mais pas espagnol. Comment pouvons-nous faire pour nous comprendre l'un l'autre ?

Peut-être que nous connaissons une langue commune. Effectivement, lui et moi parlons anglais : on va pouvoir se comprendre !]

Avec l'ordinateur c'est pareil, il comprend le binaire, moi pas, on va donc trouver un langage commun : un langage informatique que lui et moi pourrons comprendre.

Des langages informatiques, il y en a pleins mais ils ont tous la même fonction : donner des instructions.

IV. L'algorithme

Mais même si on peut communiquer avec l'ordinateur, il a sa façon à lui de penser : le binaire. 0 et 1, on peut les assimiler à « Oui » et « Non ». Alors pour expliquer à quelqu'un qui ne comprend que les oui et non, comment faire pour jouer à *Super Mario* ? Il va falloir structurer ce qu'on lui demande, être clair, précis et méthodique. Et pour cela, on utilise les algorithmes.

Derrière ce nom un peu effrayant se cache en réalité quelque chose de très commun. Un algorithme est une succession d'actions (difficile de faire plus bref comme définition).

Et le meilleur exemple d'algorithme : une recette de cuisine !

Prendre des carottes.

Les éplucher.

Les râper.

Les mettre dans un saladier.

Ajouter de la vinaigrette.

Servir.

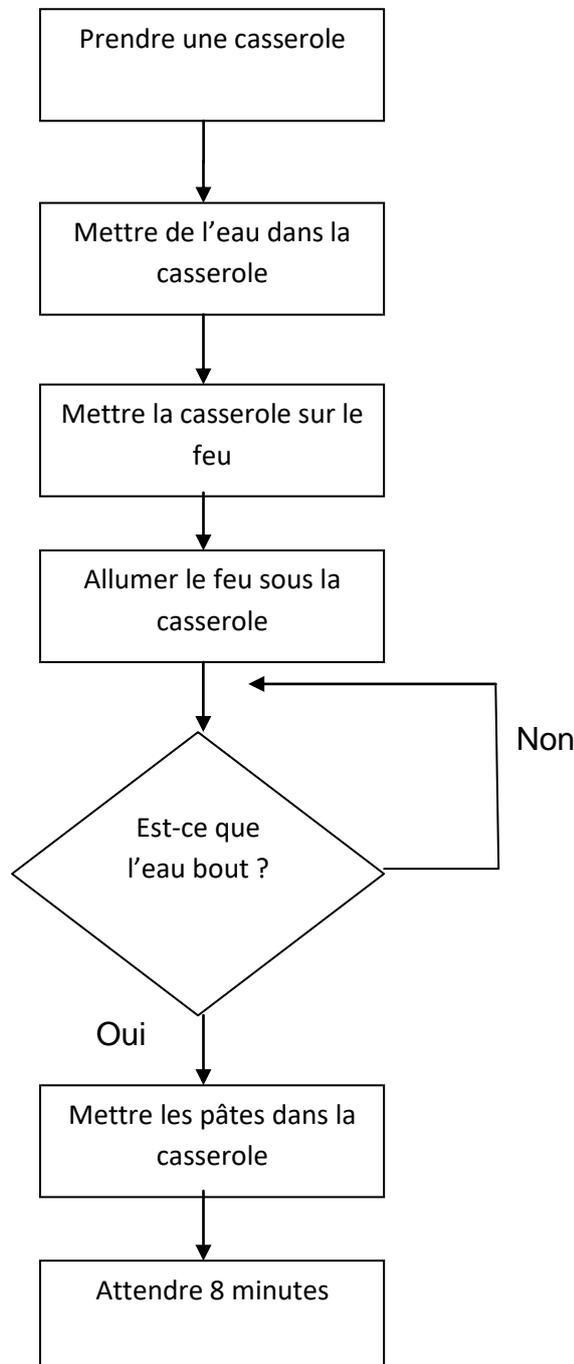
Voici l'algorithme des carottes râpées.

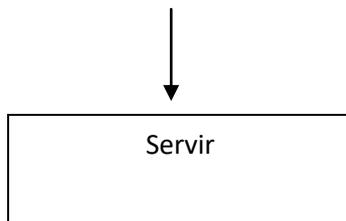
Alors vous vous doutez bien, que tous les algorithmes ne sont pas aussi simples. En réalité, il existe ce qu'on appelle des structures de contrôle qui permettent de les complexifier :

Voici les principales :

- Les conditions,
- Les boucles,
- Les variables.

L'algorithme de la cuisson des pâtes





Les conditions :

Très souvent, nos actions dépendent de plusieurs paramètres :

S'il fait beau → Je ne prends pas de pull.

Les conditions permettent à un programme de faire une action en fonction d'une ou plusieurs informations. Les termes utilisés sont « Si » et « Sinon » (« If » et « Else » en anglais). Dans le cas de notre cuisson de pâte : **SI** l'eau bout, alors je mets les pâtes.

Les conditions peuvent s'imbriquer les unes dans les autres.

S'il fait beau et s'il fait chaud → Alors je sors en Tee-shirt.

Les boucles :

En programmation, un des maîtres-mots est **optimisation**. Alors souvent plutôt que de réécrire des lignes de codes, on demande à l'ordinateur de répéter une opération en utilisant une boucle.

Une boucle peut se répéter un certain nombre de fois, ou tant qu'une condition n'est pas vérifiée.

Dans notre exemple : tant que l'eau ne bout pas, il se repose en boucle la question (« Est-ce que l'eau bout ? »).

Les variables :

Les variables sont des espaces de mémoire dans l'ordinateur dans lesquels on peut lui demander de garder une information pour nous (soit sous la forme d'un nombre soit sous la forme de mots).

Dans le cas de notre exemple : la recette ne fonctionne que pour des pâtes ayant un temps de cuisson de 8 minutes. Pour du riz longue cuisson, le programme ne fonctionne pas.

Ce que nous allons donc faire, c'est qu'au début de notre recette, nous allons créer une variable que nous appellerons « temps de cuisson » qui va dépendre de ce que l'on veut faire cuire et après avoir versé le riz dans la casserole, on va « attendre : temps de cuisson ».

