Activité (déconnectée) - Introduire les langages de programmation avec le jeu du robot

TOUTES LES FICHES

par Anissa Bouchareb • BSF Belgique

CC BY SA

PUBLIC	PARTICIPANTS	ANIMATEURS	NIVEAU	PRÉPARATION	ACTIVITÉ

Description

Cette activité est une extension de l'activité "robot idiot", qui permet de développer plus avant les notions autour des langages de programmation.

Objectifs

Langage informatique
Paradigme
Instruction

Matériel

De quoi marquer un grillage au sol (corde, ruban, craies, ect….)
De quoi marquer des objets sur le grillages (de vrais objets, les dessiner avec une craie, etc….)

Contenus utilisés

Aucun

Pré-requis

Savoir lire

WORKFLOW

1 Introduction

Cette activité vise à aborder des notions de langage de programmation de façon débranchée, sans connexion Internet. Elle est une extension de l'activité "Robot Idiot ", et s'adresse aux enfants ou à des groupes parents-enfants qui ont peu de connaissances en programmation.

2 Préparation

Familiarisez-vous avec les notions présentées ci-dessous.

Familiarisez-vous avec l'activité "jeu du robot" ; il est recommandé d'avoir déjà animé cet atelier au préalable (tant pour vous-même que pour le groupe).

Imprimez les instructions qui serviront au langage basique utilisé par les participants, par exemple à partir d'images de flèches, ou d'images de blocs scratchs. Il est possible de réutiliser du matériel servant à des ateliers robotiques (Dash, Thymio), qui font parfois appel à ce type de matériel.

3 Quelques notions théoriques

Les ordinateurs ne peuvent pas comprendre de langage humain; ils ne comprennent que le **binaire** (une façon de présenter l'information avec des 0 et des 1, grâce à l'électricité). Les humains pour leur part ne peuvent que difficilement "parler binaire" (et ce n'est pas pratique du tout). On a donc inventé des **langages de programmation**, faciles à manipuler pour les humains, et qui sont traduits en binaire pour être compris par l'ordinateur.

Un langage de programmation est donc un moyen de faire le **lien** entre le/la programmeur/euse (personne qui créé des programmes) et l'ordinateur. Il permet de donner des ordres, qu'on appelle instructions, à l'ordinateur, pour lui dire quoi faire (jouer une vidéo, ouvrir un site internet,…) et comment réagir (à un clic de souris, à une touche,…).

Les langues parlées par les humains ont évolué au fil des changements de pratique : création de nouveaux mots, mots qui disparaissent, langues qui se mélangent,… Les langages de programmation eux, ont évolué avec les **progrès** des technologies.

Différents langages informatiques permettent de faire **différentes** choses : gérer des sites internet, faire une app pour smartphone, piloter des robots, etc.

En général, les langages de programmation utilisent des mots anglais pour exprimer des choses très simples, très précises. Plus un langage est proche du binaire, plus il est de “**bas niveau**“. Plus un langage est éloigné du binaire et permet d’exprimer des concepts abstraits, plus il est de “**haut niveau**“. Dans Scratch, on peut choisir la langue humaine dans laquelle on va programmer, et il s’agit d’un langage de haut niveau.

4 Quelques exemples de langages

Les langues humaines sont regroupées en “familles” : langues germaniques, langues romanes, etc, en fonction de leurs origines et de leurs caractéristiques communes. Les langages informatiques eux, ont des **paradigmes** : des façons de fonctionner, qui permettent de programmer dans des styles différents (on représente les choses et on les manipule de façon différentes).

Scratch : il s’agit d’un langage par **blocs** qui permet d’introduire le public à la programmation. Il est simple et pratique pour s’initier à la programmation.

HTML : c’est un langage de **balises** utilisé pour créer des pages web. Il permet de coder en hypertexte.

Javascript : c’est un langage **orienté objet** qui permet de rendre des pages web dynamiques (faire des animations, des jeux, des applications en ligne,…).

Python : également orienté objet, il permet de faire plein de choses différentes et est très utilisé, y compris dans le domaine éducationnel.

C : c’est un langage **procédural** bas niveau, qui permet de manipuler des composants électroniques. Du coup, il est utilisé dans plein d’appareils, depuis le lave-vaisselle jusqu’aux satellites en passant par les voitures, les robots et les avions.

C++ : c’est l’évolution orienté objet, haut niveau du C, il permet comme Python de faire plein de choses différentes, depuis les logiciels de gestion d’entreprises aux jeux vidéos.

5 Extension du jeu du robot

Utiliser une discussion avec des questions guidées, ou une autre méthode au choix, pour aborder les **notions** ci-dessus avec le groupe. Le plus important à faire passer est qu’un langage de programmation sert à s’adresser aux ordinateurs sans devoir parler binaire, et qu’il en existe différentes sortes. Expliquer qu’à présent, le groupe

va pouvoir utiliser un langage très simple pour faire le jeu du robot en programmant.

Conserver le dispositif mis en place pour le jeu du robot. Y ajouter les instructions imprimées, et les passer en revue avec le groupe pour s'assurer que tout le monde comprend les flèches/blocs/mouvements de la même façon (attention que "pivoter" et "tourner" ne veulent pas dire la même chose).

Comme avec le jeu de base, mettre les participants par deux. L'un jouera le rôle du "programmeur" et l'autre celui du robot qui exécute le programme. Les répartir sur différents points de départ. Présenter l'histoire-prétexte et l'objectif des robots, au choix en fonction du parcours dessiné et des objets/dessins mis en place (ex: passer par le coffre puis prendre la couronne). Les "programmeurs" devront utiliser les panneaux d'**instructions**, sans parler, pour communiquer leurs commandes aux robots de façon claire et dans le langage informatique choisi.

Dans un premier temps, les programmeurs montrent **un panneau à la fois** et les robots exécutent tout de suite l'ordre.

Quand tout le monde s'en sort bien, refaire des groupes de 3 : deux programmeurs et un robots, les programmeurs discutent pour choisir les instructions nécessaires et les placer au sol, dans l'ordre. Quand ils pensent que leur **programme** ainsi créé est prêt, ils lisent les instructions au robot, qui exécute tout d'affilée. Si une erreur survient, le robot revient au début et les programmeurs doivent corriger leur programme.

6 Pour aller plus loin

Conseil médiation

Pour aller plus plus loin sur le sujet, nous vous conseillons de vous référer à la fiche outil "Algorithmes et langages de programmation"