**Formation à l’enseignement de l’informatique au collège**

**Informatique**

**Collège**

**Chapitre 3**

**Traitement des informations**

|  |  |  |
| --- | --- | --- |
|  | Connaissances |  |
| Programme de mathématiques | **Algorithmique et programmation** | |
|  | Notion d’algorithme et de programme |
|  | Notion de variable informatique |
|  | Déclenchement d’une action par un événement, séquences d’instructions, boucles, instructions conditionnelles |
| Programme de technologie | **Écrire, mettre au point et exécuter un programme** | |
|  | Notion d’algorithme et de programme |
|  | Notion de variable informatique |
|  | Déclenchement d’une action par un événement, séquences d’instructions, boucles, instructions conditionnelles |
|  | Systèmes embarqués |

|  |  |
| --- | --- |
| *Programmation graphique avec scratch* | [1 Traitement de l’information 2](#_Toc442473466)  [**1.1 Présentation 2**](#_Toc442473467)  [**1.2 Exécution d’un programme sur ordinateur 2**](#_Toc442473468)  [**1.3 Exécution d’un programme sur une cible 2**](#_Toc442473469)  [2 Structures algorithmiques 2](#_Toc442473470)  [**2.1 Blocs d’instructions 3**](#_Toc442473471)  [**2.2 Répétition – Boucle infinie 3**](#_Toc442473472)  [**2.3 Gestion événementielle élémentaire 3**](#_Toc442473473)  [**2.4 Structures conditionnelles – Si, Sinon, Alors 4**](#_Toc442473474)  [**2.5 Structures itératives – Tant que, Faire – Répéter n fois … 4**](#_Toc442473475)  [3 Fonction-Procédure 4](#_Toc442473476)  [**3.1 Procédure 4**](#_Toc442473477)  [**3.2 Fonction 5**](#_Toc442473478) |

# Traitement de l’information

## Présentation

|  |  |  |
| --- | --- | --- |
| On s’intéresse ici à la fonction « Traiter ». Celle-ci est généralement réalisée par une carte électronique, ou par le processeur de l’ordinateur, d’une tablette ou d’un téléphone. Cette fonction est essentielle car c’est elle qui permet de traiter des entrées, de commander des sorties et de communiquer avec l’utilisateur ou l’environnement. |  | |
| **Exemples :**   |  |  |  | | --- | --- | --- | |  |  |  | | *Microcontrôleur de carte*  *Arduino Uno* | *Carte mère d’une tablette « Surface »* | *Carte mère d’un ordinateur de bureau* | | |

## Exécution d’un programme sur ordinateur

Dans le cadre de l’utilisation d’un ordinateur, le processeur de la carte mère traite des informations qu’il a déjà en mémoire, en interaction avec des informations du clavier de la souris (ou de l’écran tactile). Les informations traitées peuvent être affichées sur l’écran ou diffusées par les enceintes.

|  |
| --- |
| **Exemples : Candy crush**    Dans ce cas, l’ordinateur se suffit à lui-même. L’utilisateur peut créer des programmes permettant d’atteindre un objectif sans « composants » extérieurs. |

## Exécution d’un programme sur une cible

|  |  |  |
| --- | --- | --- |
| Dans le cadre d’un système embarqué, on peut considérer :   * la phase de développement où le système est relié à l’ordinateur afin de mettre au point un programme ; * la phase d’utilisation où le programme est terminé. Il est alors « téléversé » dans la carte du système. Celui-ci peut être déconnecté de l’ordinateur et fonctionner de manière autonome. | |  | | --- | | **Exemples :** | |

# Structures algorithmiques

Afin de présenter les différentes structures algorithmiques, nous allons utiliser scratch et réaliser une implémentation du jeu « pong », l’un des premiers jeux à avoir été développé sur un ordinateur.

## Blocs d’instructions

|  |
| --- |
| **Définitions :**  **Expression**  Une expression est l’évaluation d’un calcul. Un résultat est retourné.  **Instruction**  Une instruction est une action utilisée dans un algorithme. Une instruction peut inclure une expression. Lorsqu’on écrit une suite d’instructions successives on parle de **bloc d’instructions**. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Exemples – Scratch**   |  |  | | --- | --- | |  |  | | *Instructions réalisables avec scratch* | *Bloc d’instructions* | |

|  |  |  |
| --- | --- | --- |
| **Exemple – Pong**  Une des premières étapes consiste à gérer le comportement de la balle seule.  Un mouvement de la balle correspond à un déplacement de 10 pixels. De plus, on souhaite qu’elle rebondisse quand elle rencontre un mur.   |  |  | | --- | --- | |  | Le problème est qu’il est nécessaire relancer indéfiniment le script pour que la balle évolue dans le cadre. | |

## Répétition – Boucle infinie

|  |
| --- |
| **Définition :**  Structure algorithmique permettant de répéter un nombre infini de fois des instructions. |

|  |  |  |
| --- | --- | --- |
| **Exemple – Pong**  On va maintenant faire en sorte que la balle avance « toute seule » et qu’elle rebondisse sur les bords.  Avec le premier bloc d’instructions, on se rend compte que la balle ne se déplace que de droite à gauche puis de gauche à droite indéfiniment.  Pour qu’elle couvre l’écran, il va falloir réaliser une **initialisation**.   |  |  | | --- | --- | |  |  | |

|  |
| --- |
| * **Lien à faire avec le programme de mathématiques**   + **Thème D : espace et géométrie** – (Se) repérer sur une droite graduée, dans le plan muni d’un repère orthogonal, dans un parallélépipède rectangle ou sur une sphère. |

## Gestion événementielle élémentaire

|  |
| --- |
| **Définition :**  La programmation événementielle permet de prendre en compte des événements pouvant apparaître aléatoirement et d’interférer sur le déroulement du programme.  Cela permet donc de prendre en compte un clic de souris, un appui sur une touche du clavier *etc*. |

|  |  |
| --- | --- |
| **Exemple – Pong**  On va créer un événement déclencheur pour démarrer le jeu. On va pour cela utiliser le drapeau. |  |
| Dans le cadre de l’utilisateur d’une carte électronique comme organe de traitement, les signaux émis par les capteurs peuvent être utilisés comme événement faisant évoluer le programme (exemple : capteur ultrason détectant un mur). | |

## Structures conditionnelles – Si, Sinon, Alors

|  |
| --- |
| **Définition :**  Les instructions conditionnelles permettent d’effectuer une instruction (ou un bloc d’instruction) si une condition est vraie. |

|  |  |  |
| --- | --- | --- |
| **Exemple – Pong**  On va utiliser les itérations infinies, les instructions conditionnelles et les événements pour gérer le deuxième acteur du jeu à savoir la raquette.  L’instruction conditionnelle permet aussi de gérer le mouvement de balle sur la raquette. |  |  |
| *Mouvement de la raquette* | *Mouvement de la balle sur la raquette* |

L’instruction Si « Condition » Alors « Instruction » nécessite l’emploi d’une condition dite booléenne qui peut être vraie ou fausse. Dans le cas où la condition est vraie, l’instruction située après Alors est exécutée.

|  |
| --- |
| **Définition :**  Si la condition est fausse et qu’une instruction doit être réalisée, on utilise la commande :  **Si** « Condition » **Alors** « Instruction1 »  **Sinon** « Instruction2 » |

Plusieurs conditions peuvent être imbriquées les unes dans les autres.

## Structures itératives – Tant que, Faire – Répéter n fois …

|  |
| --- |
| **Définition :**  Une instruction itérative permet de répéter un bloc d’instructions un nombre défini de fois. |

|  |
| --- |
| **Exemple** |

On peut utiliser une variable qui se modifie automatiquement à chaque itération.

# Fonction-Procédure

## Procédure

|  |
| --- |
| **Définition :**  Une procédure est une portion de code, un sous-programme, effectuant une ou plusieurs tâches indépendamment du reste du programme. |

C’est une boite noire pour celui qui utilise cette procédure : il n’est pas nécessaire de savoir ce qu’il y a à l’intérieur ou comment est faite la procédure, il faut juste savoir ce que fait cette procédure.

|  |
| --- |
| **Exemple – Clignotement d’une led**    Dans Scratch ou mBlock, on définit un Bloc. Il faut alors (zone de droite sur l’image) définir la procédure puis (zone de gauche), utiliser la procédure dans le programme principal en l’appelant par son nom. |

Une procédure peut comporter ou non des paramètres d’entrée qui sont utilisés à l’intérieur de celle-ci.

|  |
| --- |
| **Exemple – Clignotement d’une led n fois**    L’avantage d’une procédure avec arguments est de pouvoir facilement réutiliser cette procédure de manière un peu différente en fonction des arguments. |

## Fonction

|  |
| --- |
| **Définition :**  Une fonction est une procédure mais qui réalise en général un calcul et qui renvoie une ou plusieurs valeurs à la fin de son exécution. |

Aucune différence n’est faite entre procédure et fonction dans Scratch, on parle alors de blocs d’instructions avec ou non des paramètres d’entrée.