# Programme

## Caractéristiques d’un programme (définition même d’une machine/programme)

Un programme est écrit dans un fichier. Ce fichier peut être édité et lu. Le contenu peut être du texte plus ou moins compréhensible, le code source, qui est suite d’instructions écrites dans un langage de programmation. Avec un langage comme Python, ces instructions sont interprétées. Elles sont traduites ne des instructions binaires, en langage machine. Dans le cas d’un langage nécessitant une compilation, un fichier exécutable est créé. Dans la suite, aucune distinction n’est faite entre un fichier code source et un fichier exécutable.

Un programme peut être exécuté seul ou utilisé par un autre programme. Il est considéré comme une donnée par cet autre programme. Ainsi, un interpréteur ou un compilateur prend un fichier, le code source d’un programme, en argument. On peut distinguer un algorithme écrit dans un pseudo-langage d’un programme écrit dans un certain langage de programmation ; Les pseudo-langages qui sont utilisés dans la description d’algorithmes ont souvent de fortes ressemblances avec le langage de programmation utilisé ensuite pour l’implémentation. Ainsi, le langage Python peut être utilisé pour décrire des algorithmes et donc être considéré comme un pseudo-langage de description d’algorithme.

. Un fichier représentant un programme, comme un fichier représentant une image, peut être transmis, téléchargé, enregistré sur un support.

On peut considérer plusieurs programmes qui interagissent entre eux et agissent sur des données dont certaines sont d’autres programmes. C’est le cas d’un système d’exploitation instalé sur une machine.

Remarque : Un programme est exécuté par une machine. Le temps d’exécution est lié à la complexité d’un programme, son coût. La **calculabilité** permet de savoir si une fonction peut être calculée à l’aide d’un algorithme et donc si un problème peut être résolu pas une machine ou pas.

## Machine de Turing

Dans un papier publié en 1936, titré « On computable numbers, with application to the Entscheidungsproblem », (Sur les nombres calculables, avec une application au problème de la décision), Turing décrit une machine qui effectue des calculs de manière mécanique, sans intervention de l’homme à part pour l’entrée des données et la lecture des résultats. Il décrit ensuite une machine universelle capable de simuler n’importe quelle machine particulière. Cette machine abstraite imaginée par Turing comporte un ruban infini, une tête de lecture et écriture et une table de transition. Le ruban est divisé en case qui contiennent chacune un symbole d’un alphabet fini. Par défaut, les cases contiennent le symbole « blanc ». Le ruban peut se déplacer d’une case vers la gauche ou vers la droite et la tête peut lire ou écrire un symbole sur la case du ruban qui lui fait face. L’état de la machine décide du déplacement. Le nombre d’états possibles est fini et il y a un état de départ. La table de transition indique l’action à exécuter lorsque la tête lit une case du ruban en fonction de l’état courant de al machine : quel symbole écrire, comment se déplacer, et quel est le nouvel état. Cette machine représente d’une certaine manière un programme.

On peut représenter comme ci-dessous avec une double flèche qui symbolise la tête de lecture et d’écriture au-dessus du ruban, face à une case. Les symboles utilisés sont les nombres 0 et 1. La machine est par exemple dans l’état E4

E4 : ↓

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

Nous allons étudier deux exemples, les tableaux qui suivent présentent des tables de transition.

Dans la première colonne, nous trouvons les différents états et dans la deuxième colonne, ce que lit la machine. Nous trouvons ensuite ce qu’écrit la machine dans la case courante, puis le déplacement et le nouvel état. Le contenu des trois dernières colonnes dépend du contenu des deux premières.

Exemple 1 :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Etat | Lecture | Ecriture | Déplacement | Etat suivant |
| E1 | blanc | blanc | gauche | E2 |
| E2 | 0 | 1 | gauche | E2 |
| 1 | 0 | gauche | E2 |
| blanc | blanc | gauche | Fin |

La machine peut être dans deux états E1 et E2. Si la machine est dans E1 et contient un blanc, elle écrit un blanc. Le ruban se déplace vers la gauche et la machine passe dans l’état E2

Si la machine est dans l’état E2 et qu’elle lit un 0, elle écrit un 1 et si elle lit un 1, elle écrit un 0.

Dans les deux cas le ruban se déplace vers la gauche et la machine s’arrête.

Les étapes successives sont représentées ci-dessous. Initialement la machine est dans létat E1.

E1 : ↓

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

**La machine lit un blanc, donc écrit un blanc, passe à l’état E2 et le ruban se déplace vers la gauche.**

**E2 : ↓**

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

**La machine lit un 1, donc écrit un 0, reste dans l’état 2 et el ruban se déplace vers la gauche.**

**E2 : ↓**

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

**La machine lit un 0, donc écrit un 1, reste dans l’état E2 et le ruban se déplace vers la gauche.**

**E2 : ↓**

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

**La machine lit un 0, donc écrit un 1, le ruban se déplace vers la gauche.**

**E2 : ↓**

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

**La machine lit un blanc, donc écrit un blanc, le ruban se déplace vers la gauche.**

**Fin : ↓**

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

**Exemple 2 : Dans cet exemple, la machine n’a que deux états.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Etat** | **Lecture** | **Ecriture** | **Déplacement** | **Etat suivant** |
| **E1** | **blanc** | **1** | **gauche** | **E2** |
| **E2** | **blanc** | **0** | **gauche** | **E1** |

**Si la machine est dans un état E1, elle écrit un 1, le ruban se déplace vers la gauche et la machine passe dans un état E2. Si la machine est dans l’état E2, elle écrit un 0, le ruban se déplace vers la gauche et la machine passe dans l’état E1.**

**Les étapes successives sont représentées ci-dessous :**

**E1 : ↓**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

**E2 : ↓**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | **1** |  |  |  |  |  |  |  |

**E1 : ↓**

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

**E2 : ↓**

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

Le ruban étant infini, la machine écrit 10101010…, avec autant de chiffres que

…

def **machine2**(ruban, i, etat):

    while i<**len**(ruban):

        if etat==1:

            if ruban[i]==None:

                ruban[i]=1

                i+=1

                etat=2

        elif etat==2:

            if ruban[i]==None:

                ruban[i]=0

                i+=1

                etat=1

        else:

            i=**len**(ruban) *#utile s'il y a une erreur sur l'état initial*

*#Pour tester*

r=20\*[None]

**machine2**(r,0,1)

**print**(r)

[1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]

def **transition2**(ruban,i,etat):

    if etat==1:

        if ruban[i]==None:

            ruban[i]=1

    elif etat==2:

        if ruban[i]==None:

            ruban[i]=0

def **machine2**(ruban):

    i,e=0,1

    etats=[2,1]

    while i<**len**(ruban):

**transition2**(r,i,e)

        i=i+1

        e=etats[e-1]

r=20\*[None]

**machine2**(r)

**print**(r)

[1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]

Finalement, on peut imaginer que la machine n’appelle pas la fonction de transition externe à chaque étape, mais que la fonction de la transition.

def **machine2**(ruban, i, etat):

    if i<**len**(ruban):

        if etat==1:

            if ruban[i]==None:

                ruban[i]=1

**machine2**(ruban, i+1,2)

        elif etat==2:

            if ruban[i]==None:

                ruban[i]=0

**machine2**(ruban,i+1,1)

r=20\*[None]

**machine2**(r,0,1)

**print**(r)

[1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]

Lire sur internet le concept de Calculabilité et décidabilité.

P22-23 ex