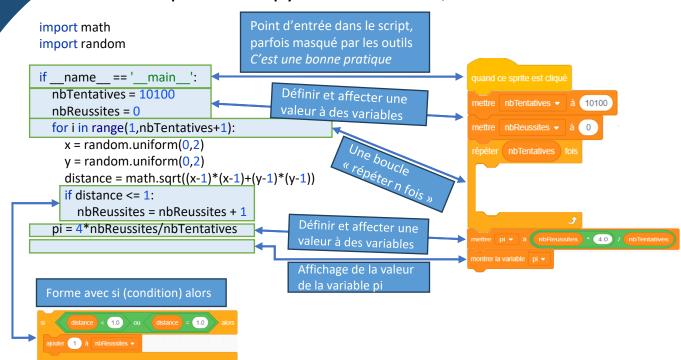
Structure générale d'un script Python

Correspondance python <-> block / scratch



Structuration du code avec les indentations

```
Niveau 0 : le « bord du fichier »
                                                                 Niveau 1 : dans le point d'entrée
   Niveau 1 : dans le point d'entrée
                                                                  nbTentatives = 10100
                                                                  nbReussites = 0
      Niveau 2: dans la boucle for
                                                                  for i in range(1,nbTentatives+1):
                                                                      # [...]
                                                                  pi = 4*nbReussites/nbTentatives
import math
                                                                  print(f"approximation de pi = {pi}")
import random
                                                                   Niveau 2: dans la boucle for
if name ==' main ':
  nbTentatives = 10100
                                                                        x = random.uniform(0,2)
  nbReussites = 0
                                                                        y = random.uniform(0,2)
 for i in range(1,nbTentatives+1):
                                                                         distance = math.sqrt((x-1)*(x-1)+(y-1)*(y-1))
    x = random.uniform(0,2)
                                                                        if distance <= 1:
    y = random.uniform(0,2)
                                                                                     # [...]
    distance = math.sqrt((x-1)*(x-1)+(y-1)*(y-1))
   if distance <= 1:
      nbReussites = nbReussites + 1
  pi = 4*nbReussites/nbTentatives
                                                       Le retour à une indentation
  print(f"approximation de pi = {pi}")
                                                       précédente met fin au « if »
```

Memo Python Version 1

Instructions et éléments de Python

Affectation: nom_variable = valeur Variable : « symbole » qui représente une valeur changeante Nom d'une variable : explicite, pour savoir ce que c'est

Il existe des mots-clés réservés par Python. Exemple: « if », « else », « for », « while », etc.

un commentaire commence par un

Type « courant » des variables :

- Nombre entier (int), Nombre « réel » (float)
- Expression booléenne (boolean, True / False)
- Chaine de caractères (string / str)

Conversion de type: int(), float(),

print : pour afficher du texte input : pour lire un texte tapé au clavier

Instruction conditionnelle « if » et les conditions

le 1^{er} n'est pas vérifié. ici on regarde si note est entre 14 et 16 if (note >= 16): print("c'est une mention TB") elif (note >= 14): print("c'est une mention B") elif (note >= 12): print("c'est une mention AB") elif (note >= 10): print("c'est une mention passable / sans mention") print("malheureusement, le bac n'est pas obtenu") pour le cas où toutes les conditions précédentes sont fausses

des booléens : True (vrai) et False (faux) et d'autres expressions entre booléen:

- a and b: (et) expression qui est vrai uniquement si a vaut vrai et si b vaut vrai - a or b: (ou) expression qui est vrai si a vaut vrai ou si b vaut vrai ou les deux
- valent vrais. - not a: (not = négation, le contraire) expression qui vaut vrai si a est faux

Les égalités / inégalités mathématiques forment des expressions booléennes : (==, !=, <, >, <=, >=)

```
Le mot-clef « while » définit
                                                 la boucle continue tant que la condition est vraie,
                   une boucle « tant que »
                                                 cela peut être une expression plus compliquée
                       while nbAllumettes > 0:
                           print(f"il reste {nbAllumettes} allumettes")
                           coupJ = int(input("Combien d'allumettes prenez-vous ? 1, 2 ou 3 : "))
Boucle while:
                          nbAllumettes = nbAllumettes - coupJ
                           print(f"Après votre tour, il reste {nbAllumettes} allumettes")
                                                                 Ce qui est répété est marqué par l'indentation
                                                                 Le contenu de la boucle peut être n'importe quelles
                  La condition doit évoluer dans la boucle, dans
                                                                 instructions (d'autres boucles, des « ifs », etc.)
                  une boucle infinie qui bloque le script
```

Le mot-clef « for » définit une boucle « pour»

Variable de boucle dont la valeur va varier entre les bornes fixées

Définition des bornes, qui vont de la 1^{re} valeur (incluse) à la 2^e valeur (exclue)

for j n range(y - 1, y + 2) y = 47192 ot pageweb = open("carte.html", "w")

Les « : » sont obligatoires

pageweb.write(f"\t\n") Boucle for : pageweb.write(f"\t\n") pageweb.write(f"\t\n")

> Ce qui est répété pour chaque valeur de la variable de boucle est marqué par l'indentation. Le contenu de la boucle peut être n'importe quelles instructions (d'autres boucles, des « ifs », etc.)

Memo Python Version 1

