

Bilan programmation Python.

Exécution programme

Processeur ← langage machine (code machine, code binaire) ← langage d'assemblage avec assembleur ← langage de programmation (python, java, C++,...)

Différentes programmations :

→ programmation impérative : exécution modifie état des données

→ programmation objet : avec classes façon d'organiser données et traitements

→ programmation fonctionnelle : se rapproche des théories maths

Différentes exécutions :

→ compilation : un compilateur traduit code en langage machine

→ interprétation : interpréteur lit prog et assure interprétation ligne à

ligne

→ byte-code : techniques où il y a mélange des deux modes

d'exécution

Langage, interpréteur

langage programmation :

Python inventé en 1991 par Guido van Rossum

des mots clés

extension '.py'

: commentaires

des modules à importer

structures de contrôle, boucles, fonctions :

indentation

blocs d'instructions

interpréteur : Pyzo, c'est un IDE (integrated development environment)

Python – constructions élémentaires

variable → une valeur

→ un nom

type de variables : entier → int (integer)
décimal → float (flottant)
caractères → str (string)
booléen → True, False (boolean)

affectation : a = 3 → on place la valeur 3 dans la variable a

opérations : +, -, *, /, //, %

fonctions : pow(a,b)
abs(a)
int(x.yz)
round(a.bcd, x)
min(a, b, c)
max(a, b, c)

import module :

- math : cos(x radian)
sin(x)
sqrt(x)
pi
floor(a.b)

- fractions : Fraction(a, b)
numerator
denominator

- random : randint()

chaînes de caractères :

eval(a)
len(chaine)
chaine[i]
chaine.count('a')
concaténation avec +
str(nb) transforme nb en chaîne de car

fonction : def
return

expressions booléens :

opérateurs : ==, <=, >=, !=, True ou False
expressions : a == b ; a ≤ b ; a != b ; a >= b

conditions : si ... sinon if ... else
si ... sinon-si ... sinon if ... elif ... else

opérateurs logiques : ou : or
et : and

boucles non bornées : while

boucles bornées : for