JAVASCRIPT

# Introduction-Les différentes classes de langages :

## A-Langage interprété :

Il y a un interpreteur entre le codeur et la machine physique, certains interpreteurs anticipent le future.

Langage lent

Les fonctions sont dynamiques

* Python/java/perl
* Shell/bash
* JS

## B-Langage Natifs :

Pas de traduction

Fonctions connues au départ

Les fonctions sont fixées.

* C, C++
* Pascal
* Go/Rust

## A-Fonctionnel :

Pas de boucles, pas de mémoires globales

* Haskell/Elm
* O-CAML
* LISP :1er langage fonctionel
  + Auto génération de code
* CLOJURE

## B-Impératif :

Il y a des boucles donc peut s’écrire avec des récursions de fonctions.

Exemple : algorithme de tris

Compilation : transformation d’un langage A vers un langage B (class java vers bytecode java (assembleur))

## A-Programmation orientée objet

Fonction privé ou publique

Il y a l’opérateur .

# II-La théorie des langages de programmation

1er compilateur : 1957

1993 : IBM coule et apparait le WEB🡪 informatique distribuée disponible pour tout le monde tout le temps

Von neuman aide à la réalisation

Hilbert

* Turing : création de la machine d’execution , ex un prg est un cgt d’état
* Church : lambda calcul🡪 un programme est une récursion de fonction
* Gödel : théorème d’accomplitude
  + Regarder des vidéos pour comprendre
  + Au niveau de l’arythmétique, il y a des théorèmes qu’on ne peut pas résoudre

Java coté serveur, machine virtuelle java

2009 : android (java)

Javascript : outil de manipulation du dom(structure de pages html), du html

Langage fonctionnel orienté web, pour le web

Discute avec le réseau, et la page

Machine virtuelle V8 dev par google (super rapide)

Node.js (js sans html)

JAVASCRIPT

* Programmeur fonctionnel plus recherché
* Langage lent
* Rapidité de création
* Ne pas chercher l’optimisation d’execution car js est lent

Diff en java/=js

Js voit des fonction dans des objets

<http://www.modulecounts.com/> 🡪 compte le nombre de module créés par langage🡪 bilan haskell on oublie

on ne sait pas quand l’evenet arrive

programmation par rapport aux evenements

système asynchrone

expression régulière

# Promesses

En interne, 3 états possibles :

accomplie, rejetée, en attente