

# Architecture Logicielle et Matérielle des Ordinateurs v2 (ALMO2)

1. Planning
2. Objectifs du module
3. Prérequis

## Planning

- Ce module ouvrira en septembre 2020

## Objectifs du module

- Le but de ce module est d'expliquer en détails comment plusieurs programmes peuvent s'exécuter indépendamment sur un ordinateur multicores à mémoire virtuelle.
- Le module introduit **progressivement**
  - ◆ les composants matériels
  - ◆ et les services du système d'exploitation.
- L'architecture matérielle est émulée au bit et au cycle près pour une exécution du code depuis le signal reset (full system emulation).
- Il y a un TP par semaine :
  - ◆ Le premier code exécuté est un simple programme en assembleur affichant "Hello World !" sur un terminal sans aucun système d'exploitation.
  - ◆ Le dernier code exécuté est constitué de deux programmes indépendants s'exécutant chacun dans leur espace de mémoire virtuelle au-dessus d'un système d'exploitation POSIX like.
- Le module doit permettre de répondre aux questions suivantes (liste non exhaustive) :
  - ◆ Comment démarre une machine ?
  - ◆ Quel est le rôle d'un système d'exploitation et quels services propose-t-il ?
  - ◆ Qu'est-ce qu'un cache de mémoire ?
  - ◆ Qu'est-ce qu'un contrôleur de périphérique, et comment l'utiliser ?
  - ◆ Qu'est-ce qu'un pilote de périphérique ?
  - ◆ Comment exécuter plusieurs programmes sur un seul processeur ?
  - ◆ Comment plusieurs processeurs peuvent-ils se partager le même espace d'adressage ?
  - ◆ Pourquoi chaque programme doit avoir un espace d'adressage propre et comment ?
  - ◆ Lorsqu'on frappe une touche du clavier, comment le caractère s'affiche-t-il à l'écran ?

## Prérequis

- Une bonne connaissance du langage C est nécessaire puisque le module propose d'écrire un système d'exploitation depuis le début.
- Une connaissance de l'assembleur MIPS32 est aussi importante, mais les principes de ce langage et les conventions d'usage pour l'écriture des fonctions sont rappelés la première semaine.