

## LO22-AI20 – TP2

### Arbres de défaillances

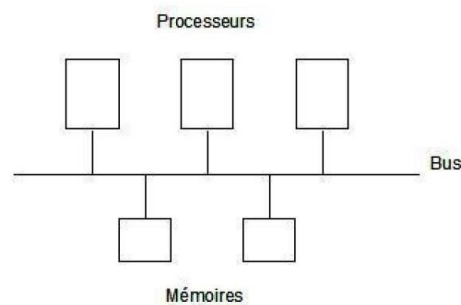
**Objectif** : Evaluer les paramètres de sûreté de fonctionnement (fiabilité, disponibilité, etc.) d'un système informatique à l'aide de la méthode des arbres de défaillance, en utilisant <http://fiabilipy.org/>.

#### Présentation du module « FT »

Pour obtenir les résultats de fiabilité et disponibilité d'un système, il est nécessaire de combiner les données de fiabilité de ses composants selon l'architecture fonctionnelle et/ou matérielle du système. L'outil *fiabilipy* permet la modélisation de la logique dysfonctionnelle d'un système sous forme d'un arbre de défaillance ainsi que les liens entre ses composants. Ensuite, l'outil permet le calcul de la fiabilité et de la disponibilité du système complet obtenue à partir des données de fiabilité des composants.

### 1. Etude de la fiabilité et de la disponibilité d'un système informatique

On considère un système informatique formé de deux processeurs, d'une mémoire et d'un bus de communication. Le fonctionnement nominal du système nécessite que les deux processeurs soient non défaillants ainsi que la mémoire et le bus de communication. Pour améliorer la fiabilité du système, on a prévu d'utiliser trois processeurs P1, P2 et P3 et deux mémoires M1 et M2 (cf. figure ci-dessous).



Les données de fiabilité des composants sont données dans le tableau suivant :

Composants	Taux de défaillance $\lambda$ par heure
Processeurs	$2.28 \times 10^{-4}$
Mémoires	$2.94 \times 10^{-4}$
Bus	$10^{-4}$

Hypothèses :

- Les probabilités de défaillances des composants suivent des lois exponentielles.
  - Il n'existe pas de causes communes de défaillance.
  - Les composants sont non réparables.
- 1.1. Donner l'arbre de défaillance dont l'événement sommet S représente la défaillance du système informatique.
  - 1.2. Tracer la disponibilité du système informatique sur une période de 5 ans. Commentez les résultats obtenus.
  - 1.3. A l'aide d'un calcul du facteur d'importance de Birnbaum de chaque composant, identifiez le maillon faible du système.
  - 1.4. Proposer une solution pour améliorer la fiabilité du système.
  - 1.5. Refaire les questions 1, 2 et 3 en rajoutant au système étudié :
    - 1.5.1. Une source d'alimentation -> architecture 1.
    - 1.5.2. Deux sources d'alimentation -> architecture 2.
    - 1.5.3. Trois sources d'alimentation avec un vote de 2/3 -> architecture 3 (*fiabilipy* ne peut pas créer un système avec deux voteurs).Taux de défaillance d'alimentation est  $2.28 \times 10^{-4}$