

## Edmond La Chance, Ph.D

edmondlachance.com

github.com/mitchi

[Edmond.Lachance@uqac.ca](mailto:Edmond.Lachance@uqac.ca)

## ÉDUCATION

**2013-2021** Doctorat en vérification logicielle et calcul distribué, Université du Québec à Chicoutimi

*Extended Combinatorial Testing using Graph Algorithms and Apache Spark*

Cette thèse présente une généralisation des tests t-way, des réductions aux problèmes de coloration de graphes et de couverture par ensembles et des algorithmes distribués utilisant Apache Spark. Les résultats expérimentaux ont été obtenus en utilisant des clusters d'ordinateurs fournis par Calcul Canada.

**2011-2013** Maîtrise en Informatique, Université du Québec à Chicoutimi

*Algorithmes pour l'arbre couvrant minimal*

Ce mémoire de maîtrise compare empiriquement différentes manières de résoudre le problème de l'arbre couvrant minimal. Plusieurs structures de données et algorithmes sont implémentés en C++ et mesurés : Union-Find, tas binomial, tas binaire, tas de Fibonacci, Algorithme de Prim, Algorithme de Boruvka, Algorithme de Kruskal.

**2008-2011** BAC en Informatique, Université du Québec à Chicoutimi

## EXPÉRIENCE PROFESSIONNELLE

**Programmeur Analyste chez Timesphere**

**Août 2021 – Maintenant**

Je travaille comme programmeur Full-Stack sur l'application Timesphere dans l'équipe SAQ2 (Société des alcools du Québec). Je travaille dans une petite équipe sur la migration de l'application Timesphere vers une nouvelle interface qui est faite avec Typescript, React et Redux-toolkit. Mon travail consiste à corriger des bogues, développer de nouvelles fonctionnalités dans notre backend Java et migrer des écrans de travail vers la nouvelle interface. C'est donc un travail Full-Stack. Toute l'équipe travaille en développement Agile.

Technologies utilisées : Typescript, React, Redux-toolkit, Java avec Spring Framework.

**Professeur chargé de cours, Université du Québec à Chicoutimi**

Cours enseignés:

### **8INF803 (Bases de données réparties)**

**2015-2021**

Dans le cours 8INF803, j'enseigne aux étudiants les bases de données réparties et le traitement de données distribué. Le cours montre de nombreuses structures de données utiles pour les applications gourmandes en données telles que les B+Trees, LSM, Bloom Filters, HyperLogLog, les index Bitmap, etc. Du côté de la programmation, nous utilisons une variété de technologies, mais principalement Apache Spark avec Scala, Java et Python. Pour les bases de données, nous utilisons MySQL, SQLite et MongoDB. Finalement, je montre à mes étudiants comment implémenter des algorithmes itératifs distribués avec Apache Spark tout en optimisant la performance.

### **8GIF128 (Conception et programmation de sites Web)**

**2015**

Dans 8GIF128, j'enseigne les bases de la programmation web en montrant HTML, CSS et Javascript (pour le côté navigateur). Du côté serveur, je montre comment utiliser Javascript avec Node.JS, Express.IO et Socket.io. Le projet final consiste à créer un site web, et plusieurs services qui communiquent avec ce dernier avec la technologie WebSockets.

## **PROJETS**

### **TSPARK – Un générateur de tests combinatoires distribué**

<https://github.com/mitchi/TSPARK>

TSPARK est un projet open source développé pour ma thèse de doctorat. TSPARK est écrit en Scala, en utilisant Apache Spark, et compte environ 30 000 lignes de code.

TSPARK est un outil qui génère des tests combinatoires (une technique pour trouver des bugs dans un logiciel). Pour générer ses tests combinatoires, TSPARK reçoit un modèle des paramètres et valeurs utilisées par le programme, avec une valeur appelée la force d'interaction, et génère ensuite une suite de tests en utilisant l'un de ses algorithmes.

TSPARK contient des algorithmes distribués pour résoudre les problèmes de coloration de graphes et de sommets d'hypergraphes. Il contient également un algorithme hybride appelé Distributed IPOG. Chaque algorithme a été testé pour optimiser ses performances. TSPARK utilise à l'interne des tableaux de bits et des tableaux de bits compressés pour optimiser ses structures de données. Finalement, TSPARK est distribué comme un fichier .jar autonome qui peut être ensuite exécuté sur un ordinateur classique, ou un cluster d'ordinateurs.

### **CTL Simulator**

CTL Simulator est un petit projet (environ 1000 lignes de C++) qui implémente l'algorithme d'étiquetage CTL pour les vérificateurs de modèles.

## **ASM FORUM**

ASM Forum est un vieux projet qui implémente un forum de discussion simple avec des fils de discussions/réponses et des utilisateurs uniques. Le forum est programmé en utilisant le langage assembleur x86, les processus CGI et SQLite pour le stockage.

## **COMPÉTENCES**

**Apache Spark**

**Java**

**Scala**

**C/C++**

**SQL**

**Slurm**

**Spring Boot**

**Node.js / Socket.io**

**Typescript, React et Redux Toolkit**

**JavaScript**

**JIRA, Git et Github**

**HTML et CSS**

**LaTeX**

**Langage assembleur x86**

