ABSTRAK

Banyak algoritma kriptografi diasosiasikan dengan *high-performace CPU*, didesain kurang memperhatikan dan memberikan apresiasi penuh terhadap performansi pada implementasinya pada tingkat *low-level software* dan diatas mikroprosesor.

Secara logik, untuk mencapai performansi yang lebih optimal pada implementasi tingkat *low level software* dalam menyelesaikan suatu masalah terkomputerisasi, solusi yang terbaik adalah dengan menggunakan sistem perangkat keras dan sistem operasi yang lebih canggih dan lebih baik, atau menggunakan bahasa pemrograman tingkat rendah seperti *assembler*.

Dengan mengesampingkan dan mengasumsikan bahwa sulit untuk merealisasikan hal-hal solusi optimasi diatas, maka pada optimasi dari implementasi algoritma-algoritma kriptografi yang banyak dilandasi teori-teori dan rumus matematik, diperlukan teknik-teknik dan metode-metode optimasi yang lebih efisien yang mampu menyederhanakan kompleksitas dan komputasi algoritma dengan harapan mampu menghasilkan performansi waktu pemrosesan dan performansi *throughput* yang lebih baik.

Java™ sebagai bahasa pemrograman yang identik dengan *Java Virtual Machine* (JVM) nya dan citra negatif sebagai bahasa yang lambat, ditambah dengan implementasi algoritma-algoritma kriptografi yang tidak dioptimasi semakin melekatkan bahasa pemrograman Java dengan citra lambat diatas.

Untuk dapat memenuhi performansi yang lebih baik yang dilandasi oleh dua paragraf diatas, dalam Tugas Akhir ini dikaji, diimplementasikan dan dianalisis teknik-teknik optimasi algoritma pada kasus implementasi algoritma-algoritma kriptografi berbasis Java, yang kemudian diharapkan dapat dihasilkan fokus teknik-teknik optimasi yang lebih sempit dan peningkatan performansi yang lebih baik.