

## ABSTRAK

Program aplikasi biasanya ditulis dalam bahasa tingkat tinggi. Karena batasan kemampuannya, sulit untuk menghasilkan program yang benar-benar optimal pada bahasa tingkat tinggi. Misalnya untuk mendapatkan elemen suatu array berdimensi dua dalam bahasa Pascal, hanya bisa dilakukan dengan menuliskan  $A[i,j]$ . Pada pelaksanaannya, elemen array tersebut diakses dengan serangkaian penghitungan lokasi yang tentunya relatif membutuhkan waktu. Dalam tingkat *intermediate code*, penghitungan lokasi dapat dilihat dengan jelas sehingga memungkinkan dilakukannya penyederhanaan perhitungan yang akan memperbaiki waktu pelaksanaan.

Ada beberapa hal yang ingin dicapai dalam mengerjakan Tugas Akhir ini. Pertama, untuk mengimplementasikan teknik-teknik optimasi *intermediate code* pada kompilator dengan grammar Bahasa Indonesia sehingga dapat menghasilkan *intermediate code* yang lebih baik. Kedua, untuk mengamati perubahan *intermediate code* yang terjadi akibat proses optimasi. Ketiga, untuk mengetahui pengaruh optimasi *intermediate code* di dalam pelaksanaan program.

Optimasi *intermediate code* merupakan usaha untuk mengubah *intermediate code* menjadi lebih efisien, baik dilihat dari waktu pelaksanaan maupun besarnya pemakaian memori. *Intermediate code* merupakan suatu representasi internal program sumber.

Pada prinsipnya, proses optimasi *intermediate code* dijalankan dalam tiga tahap. Pertama adalah tahap analisa aliran kendali. Ditandai dengan pengidentifikasian blok dasar dan pembentukan graf alir. Kedua adalah tahap analisa aliran data. Dalam tahap ini dikumpulkan berbagai informasi mengenai keadaan program dengan melihat bentuk aliran data yang sudah didapatkan dari tahap sebelumnya. Tahap ketiga adalah melakukan perubahan-perubahan (penambahan, pemindahan atau pengurangan) *intermediate code*, berdasarkan informasi yang sudah dikumpulkan. Perubahan-perubahan *intermediate code* yang bisa dilakukan dalam optimasi antara lain : eliminasi sub-ekspresi umum, perintah penyalinan (*copy statement*), pemindahan perintah loop-invarian, reduksi strength, dan eliminasi variabel induksi.

Obyek optimasi *intermediate code* adalah *intermediate code* yang dibangkitkan kompilator bahasa pemrograman dengan grammar Bahasa Indonesia. Input dari optimasi *intermediate code* adalah program sumber (*source program*), sedang hasil dari proses optimasi adalah file *intermediate code* yang sudah dioptimasi dan file output yang berisi proses optimasi dari awal sampai akhir.