

## ABSTRAK

Agar dapat menggunakan komputer dengan baik, orang harus mempelajari instruksi-instruksi (*command*) dan bahasa-bahasa komputer tertentu. Hal ini membutuhkan waktu dan latihan. Inilah yang menjadi alasan utama mengapa komputer kurang dapat memasyarakat. Penggunaan komputer akan lebih mudah jika orang dapat berkomunikasi dengan komputer dalam bahasa sehari-hari. Pengolahan bahasa alamiah, yang pada dunia komputer lebih dikenal dengan *natural language processing*, akan membuka kemungkinan bagi manusia untuk melakukan dialog dengan komputer dalam bahasa sehari-hari. Program pengolah bahasa alamiah akan membuat komputer dapat menganalisa, memahami, serta memberikan tanggapan.

Skripsi ini mencoba menerapkan konsep-konsep pengolahan bahasa alamiah untuk membuat sebuah program prototipe pengolahan bahasa alamiah dalam tata bahasa Indonesia. Kasus yang diambil adalah sebuah proses *query* terhadap obyek barang dan harga. Disamping itu pada program dapat melakukan proses *query*, program juga ditambahkan fasilitas untuk menjalankan beberapa perintah *DOS*.

Dalam perancangan program ini, pertama-tama yang harus dikerjakan adalah mendefinisikan bahasa agar dimengerti oleh program. Pendefinisian meliputi pendefinisian tata bahasa, perbendaharaan kata, serta kelompok-kelompok katanya. Tata bahasa Indonesia diterjemahkan kedalam bentuk matematis sesuai teori bahasa formal.

Langkah berikutnya adalah merancang masing-masing elemen sistem pengolahan bahasa alamiah, yaitu *lexicon*, *knowledge base*, *scanner*, *parser*, *understander*, dan *generator*. *Lexicon* dan *knowledge base* berbentuk basis data. Basis data *lexicon* memuat segala perbendaharaan kata yang dikenal oleh program. *Knowledge base* memuat data yang sebenarnya, yaitu barang dan harga. Empat elemen yang lain berupa modul program. *Scanner* dibuat untuk menganalisa kalimat input secara leksikal. Sedangkan *parser* melakukan analisa kalimat secara sintaktik dan semantik. Tata bahasa dalam bahasa formal yang telah didefinisikan sebelumnya akan ditempatkan pada modul *parser* ini. Jenis *query* dan perintah juga didefinisikan pada bagian *parser*. *Understander* adalah bagian yang menjalankan proses *query* atau perintah yang telah ditentukan oleh bagian *parser*. Bagian yang terakhir, *generator*, bertugas menampilkan hasil sesuai dengan proses yang dikerjakan oleh *understander*.