

PERTEMUAN 1 :

KONSEP TEORI OTOMATA (AUTOMATA) DALAM TEORI KOMPUTASI

A. TUJUAN PEMBELAJARAN

Pada bab ini akan dijelaskan definisi tentang teori bahasa dan otomata yang akan selanjutnya beberapa notasi yang berkaitan dengan perkuliahan. Setelah menyelesaikan perkuliahan, mahasiswa diharapkan mampu : Menjelaskan kedudukan konsep teori otomata dalam teori komputasi.

B. URAIAN MATERI

Para peneliti mulai studi mengenai mesin abstrak¹ sederhana yang kenal dengan “Otomata” ini di tahun 1940-an dan 1950-an. Seorang ahli bahasa Noam Chomsky memulai mempelajari tata bahasa formal di tahun 1950-an. Meskipun bukan mesin, namun tata bahasa ini berhubungan langsung dengan mesin abstrak atau automata dan saat ini merupakan dasar dari komponen software² (perangkat lunak), termasuk dengan kompilernya.

Sedangkan S.Cook mendalami teori dari hasil penelitian Turing tentang apa yang dapat dan tidak dapat diproses oleh komputer. Cook mampu memisahkan antara masalah – masalah yang kemudian dapat menyelesaikan secara efisien, walaupun membutuhkan waktu yang cukup lama dalam prakteknya.

Teori – teori pengembangan bahasa berkaitan sekali dengan ilmu komputer kini. Bahkan beberapa konsep, seperti otomata, teori bahasa formal tertentu digunakan sebagai dasar desain beberapa software – software yang penting seperti pada kompilernya pada bahasa pemrograman, mesin pencari (search engine) dan masih banyak lagi contoh implementasi dari teori bahasa dan otomata. Selanjutnya teori mesin Turing, berimplikasi untuk mengenali apa harapan yang diperoleh dari pengembangan software seperti menentukan metode untuk mengatur jumlah waktu yang dibutuhkan program untuk menyelesaikan suatu masalah.

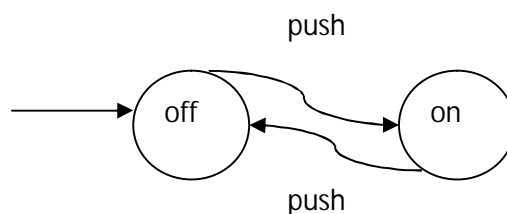
¹ Istilah mesin abstrak disini adalah bukan mesin fisik yang sesungguhnya melainkan model yang dikembangkan menjadi mesin yang mampu mengakomodir input, proses hingga output

² Baik perangkat lunak maupun penunjangnya

Sekarang mari lihat apakah teori otomata dari sisi kegunaan nya seperti yang sebelumnya dikatakan pada bab awal ini bahwa Otomata merupakan model yang sangat penting dalam pengembangan hardware maupun software menurut J. Hopcroft dalam buku nya, yakni :

- 1) Aplikasi untuk mendisain dan melakukan pengecekan dari pergerakan rutin suatu sirkuit digital
- 2) Analisa Leksikal dari tipe kompiler, yakni komponen kompiler yang menerjemahkan teks input menjadi unit – unit logika, seperti identifier³, keywords⁴ dan punctuation⁵.
- 3) Aplikasi untuk melakukan scanning atau mengecek seluruh bagian banyak teks, seperti kumpulan halaman website, untuk menemukan kebutuhan menyesuaikan kata, frasa kata atau pola lainnya.

Contoh 1 : misalkan sebuah nontrivial mesin abstrak otomata adalah sebuah pilihan tombol on/off. Alat tersebut mengingat akan posisi dalam keadaan “on” ataupun posisi dalam keadaan “off” dan pengguna nya boleh untuk menekan tombol tersebut dengan hasil yang berbeda, tergantung dari tombol yang ditekan. Ketika posisi tekan dalam suatu keadaan tertentu , kemudian menekan tombol akan merubah menjadi posisi ke keadaan tertentu dan jika tombol dalam keadaan off maka dengan menekan tombol akan merubahnya menjadi keadaan on dan sebaliknya.



Gambar 1.1 Contoh Model Otomata sebuah tombol on/off

³ Istilah *identifier* adalah suatu pengenalan atau pengidentifikasi yang dideklarasikan agar selanjutnya dikenali oleh kompiler. Berupa nama variabel, konstanta fungsi, kelas, *template* maupun *namespace*

⁴ *Keyword* adalah kata kunci yang dikenal oleh kompiler sebagai bentuk yang konstan dan tidak boleh digunakan sebagai *identifier*

⁵ Punctuation adalah tanda baca yang juga dikenali dalam kompiler yang memiliki fungsi masing-masing dalam program

Teori bahasa yang akan dibicarakan dalam Teori Bahasa dan Otomata (Automata) ini merupakan suatu bahasa formal (formal language), diutamakan untuk perancangan kompilator (compiler) dan sebagai pengolah proses naskah (text processor). Bahasa formal merupakan kumpulan suatu kalimat dan kalimat dalam sebuah bahasa berasal dari sebuah tata bahasa (grammar) yang sama. Sebuah bahasa formal bisa dibangun dari dua atau lebih tata bahasa berbeda. Dikatakan bahasa formal karena tata bahasa dibuat melalui pembangkit setiap kalimatnya. Bahasa Natural / manusia bersifat sebaliknya, tata bahasa dibentuk untuk standarisasi kata-kata yang ada di lingkungan masyarakat. Selanjutnya 'bahasa formal' akan disebut 'bahasa' saja.

Teori Otomata (Automata) adalah pembelajaran mengenai mesin yang abstrak berbasis komputasi. Sebelum adanya komputer di tahun 1930-an, Turing mempelajari sebuah mesin abstrak yang mampu melakukan semua yang dapat dilakukan oleh komputer sekarang ini, setidaknya bagaimana komputer dapat melakukan proses hitung. Tujuan Turing adalah untuk mengetahui batasan yang ada antara yang dapat dilakukan ataupun yang tidak dapat dilakukan oleh sebuah mesin komputer. Hasil kesimpulannya tidak hanya diaplikasikan pada abstraksinya berupa "Turing Machine", akan tetapi juga diaplikasikan pada mesin nyata sekarang ini.

Menurut John Hopcroft dalam bukunya Introduction to Automata Theory Languages and Computation, otomata adalah sebuah pemodelan yang berguna untuk banyak kepentingan akan pengembangan hardware dan software. Bisa dikatakan sebagai model mesin abstrak yang dapat mengenali (recognize), menerima (accept), atau membangkitkan (generate) sebuah kalimat dalam bahasa tertentu.

C. SOAL LATIHAN DAN TUGAS

1. Silakan cari referensi mengenai konsep dasar teori bahasa dan otomata dan lakukan review
2. Berikan pendapat Anda dalam review tersebut.

D. DAFTAR PUSTAKA

Hopcroft, John. E., etc. 2001. Second edition. Introduction to Automata Theory, Languages, and Computation. US America : Pearson

Goswami, D and Krishna, K.V. November 5 2010. Formal Languages and Automata Theory.

Utdirartamo, Firrar. 2005. Edisi kedua. Teori Bahasa dan Otomata. Yogyakarta : Penerbit Graha Ilmu