

JFLAP SEBAGAI ALAT BANTU PENGAJARAN AUTOMATA

Anis Zubair

Fakultas Teknologi Informasi Universitas Merdeka Malang
anis.zubair@unmer.ac.id

Abstract

Automata is a plural form of automaton. Automaton means machine, so automata means machines. There are many kinds of automata. Automata which we discuss in this paper is DFA, NFA, CFG, and PDA. We give each of them definition, example, accepted or rejected strings, and process about how to accept string visually in JFLAP.

Keywords : JFLAP, automata, DFA, NFA, CFG, PDA

1. Pendahuluan

Automata adalah salah satu mata kuliah wajib pada jurusan atau program studi ilmu komputer, teknik informatika, atau sejenisnya. Tidak banyak perangkat lunak yang dapat digunakan untuk membantu memahami mata kuliah ini. Salah satu perangkat lunak yang dapat digunakan untuk membantu memahami mata kuliah ini adalah JFLAP.

JFLAP atau *Java Formal Languages and Automata Package* adalah perangkat lunak yang diciptakan oleh Susan Rodger dari Duke University pada sekitar 1990an. Perangkat lunak ini ditujukan untuk membantu mahasiswa untuk belajar automata. Perangkat lunak ini juga ditujukan sebagai alat bantu pengajaran automata. Perangkat lunak ini mengalami evolusi dan perbaikan selama bertahun-tahun. Saat ini keluaran terakhir adalah JFLAP 8.0.

Artikel ini menguraikan tentang peran JFLAP sebagai alat bantu pengajaran automata. Sebelumnya artikel ini mengenalkan konsep-konsep dasar automata. Konsep-konsep dasar itu adalah alfabet dan *string*. Selanjutnya jenis-jenis automata yang dibahas adalah DFA, NFA, CFG, dan PDA.

2. Automata

Istilah automata kali pertama digunakan pada makalah yang berjudul *The Logical and General Theory of Automata*. Makalah ini ditulis oleh John Von Neumann dari The Institute for Advanced Study dan disajikan di Hixon Symposium 20 September 1948 di Pasadena, California.

Sebagai kelanjutan makalah yang pernah ditulis, John Von Neumann menulis buku yang berjudul *Theory of Self-Reproducing Automata*. Buku ini diterbitkan oleh University Illinois Press pada tahun 1966.

3. Alfabet dan String

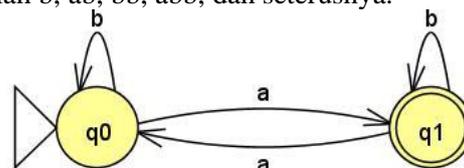
Alfabet adalah himpunan simbol yang berhingga dan tidak kosong. Sebagai contoh $U = \{a, h, i, m, s, w\}$ adalah alfabet U yang memiliki simbol $a, h, i, m, s,$ dan w .

String adalah barisan simbol yang terhingga. Simbol-simbol ini harus ada di dalam alfabet. Sebagai contoh maha, mahasiswa, mahasiswi, dan seterusnya adalah *string* yang dapat dibuat dari alfabet U .

4. DFA

DFA merupakan singkatan dari *Deterministic Finite Automata*. Komponen-komponen DFA terdiri atas *state*, alfabet, transisi, *state* awal, dan *state* akhir. Di dalam DFA *state* dinyatakan dalam bentuk himpunan, alfabet dinyatakan dalam bentuk himpunan, transisi dinyatakan dalam bentuk fungsi, *state* awal dinyatakan dalam bentuk anggota himpunan, dan *state* akhir dinyatakan dalam bentuk himpunan. DFA hanya memiliki satu transisi untuk setiap simbol pada suatu *state*.

Sebagai contoh, DFA yang dapat menerima *string* yang berakhiran b dengan alfabet yang memuat huruf a dan b dinyatakan dalam bentuk Gambar 1. *String-string* yang dapat diterima adalah $b, ab, bb, abb,$ dan seterusnya.



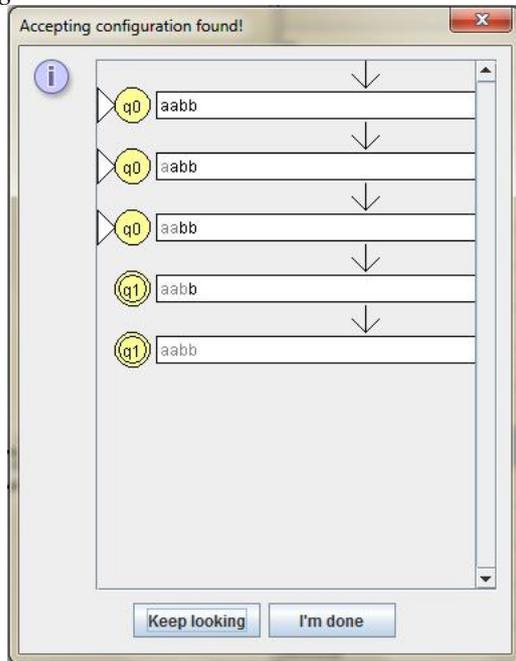
Gambar 1 Contoh DFA

Gambar 2 menunjukkan *string-string* yang dapat diterima atau ditolak. *String-string* yang dapat diterima adalah yang berakhiran b , sedangkan *string-string* yang berakhiran a ditolak.

Input	Result
a	Reject
b	Accept
aa	Reject
bb	Accept
aaa	Reject
bbb	Accept
aba	Reject
baa	Reject
bba	Reject
aaaa	Reject
bbbb	Accept
bbba	Reject
baaa	Reject
aaba	Reject
abaa	Reject
abba	Reject
baba	Reject
bbaa	Reject

Gambar 2 String-string yang Dapat Diterima atau Ditolak oleh DFA

Gambar 3 menunjukkan proses yang dilakukan oleh DFA untuk menerima suatu string.



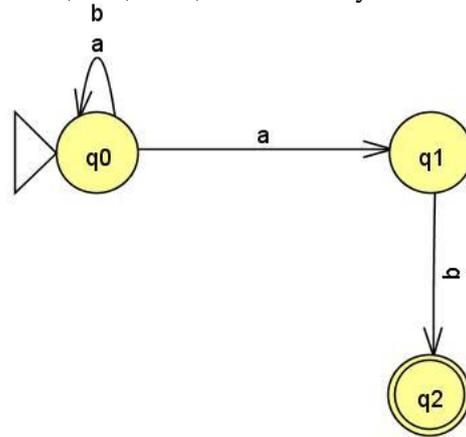
Gambar 3 Proses yang Dilakukan oleh DFA untuk Menerima Suatu String

5. NFA

NFA merupakan singkatan dari Nondeterministic Finite Automata. Komponen-komponen NFA terdiri atas state, alfabet, transisi, state awal, dan state akhir. Di dalam NFA state dinyatakan dalam bentuk himpunan, alfabet dinyatakan dalam bentuk himpunan,

transisi dinyatakan dalam bentuk fungsi, state awal dinyatakan dalam bentuk anggota himpunan, dan state akhir dinyatakan dalam bentuk himpunan. NFA bisa tidak memiliki, memiliki satu, atau memiliki lebih daripada satu transisi untuk setiap simbol pada suatu state.

Sebagai contoh, NFA yang dapat menerima string yang berakhiran ab dengan alfabet yang memuat huruf a dan b dinyatakan dalam bentuk Gambar 4. String-string yang dapat diterima adalah ab, bab, baab, dan seterusnya.



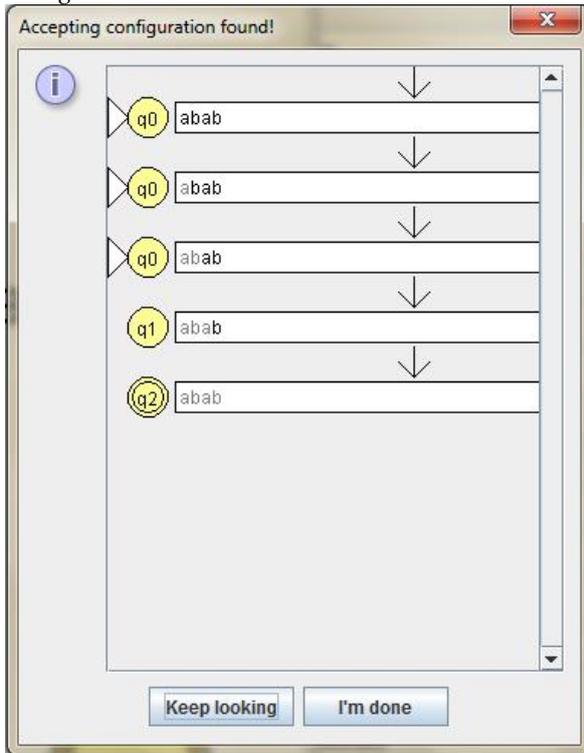
Gambar 4 Contoh NFA

Gambar 5 menunjukkan string-string yang dapat diterima atau ditolak. String-string yang dapat diterima adalah yang berakhiran ab, sedangkan string-string yang berakhiran selain ba ditolak.

Input	Result
a	Reject
b	Reject
aa	Reject
ab	Accept
aaa	Reject
bbb	Reject
aab	Accept
aba	Reject
bab	Accept
aaaa	Reject
bbbb	Reject
bbab	Accept
abaa	Reject
aaab	Accept
aaba	Reject
abab	Accept
abab	Accept
baba	Reject

Gambar 5 String-string yang Dapat Diterima atau Ditolak oleh NFA

Gambar 6 menunjukkan proses yang dilakukan oleh DFA untuk menerima suatu string.



Gambar 6 Proses yang Dilakukan oleh NFA untuk Menerima Suatu String

6. CFG

CFG merupakan singkatan dari *Context-Free Grammar*. Komponen-komponen CFG terdiri atas alfabet, terminal, aturan, dan start. Di dalam CFG alfabet dinyatakan dalam bentuk himpunan, terminal dinyatakan dalam bentuk himpunan, aturan dinyatakan dalam bentuk himpunan dan start dinyatakan dalam bentuk anggota himpunan.

Sebagai contoh, Gambar 7 adalah CFG yang digunakan untuk menguji keseimbangan kurung buka dan kurung tutup.

LHS	RHS
S	→ [S]
S	→ [A]
S	→ SS
A	→ ε

Gambar 7 Contoh CFG

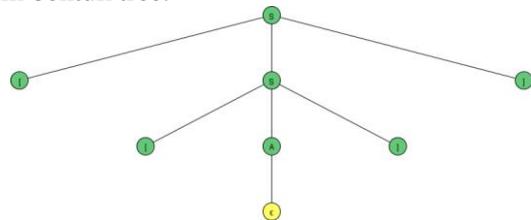
Gambar 8 menunjukkan *string-string* yang dapat diterima atau ditolak. *String-string* yang

dapat diterima adalah yang memiliki keseimbangan antara kurung buka dan kurung tutup, sedangkan *string-string* ditolak adalah yang memiliki keseimbangan antara kurung buka dan kurung tutup.

Input	Result
[]	Accept
[[]	Reject
][[]	Reject
[[[]	Reject
[[[]]	Accept
[[[]]]	Accept

Gambar 8 *String-string* yang Dapat Diterima atau Ditolak oleh CFG

Gambar 9 menunjukkan proses yang dilakukan oleh CFG untuk menerima suatu string dalam bentuk tree.



Gambar 9 Proses yang Dilakukan oleh CFG untuk Menerima Suatu String dalam Bentuk Tree

Gambar 10 menunjukkan proses yang dilakukan oleh CFG untuk menerima suatu string dalam bentuk tabel derivasi.

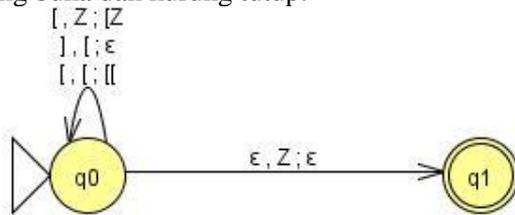
Production	Derivation
S → [S]	S
S → [A]	[S]
A → ε	[[A]]

Gambar 10 Proses yang Dilakukan oleh CFG untuk Menerima Suatu String dalam Bentuk Tabel Derivasi

7. PDA

PDA merupakan singkatan dari *Push Down Automata*. Komponen-komponen PDA terdiri atas *state*, alfabet masukan, alfabet stack, transisi, *state* awal, dan *state* akhir. Di dalam PDA *state* dinyatakan dalam bentuk himpunan, alfabet masukan dinyatakan dalam bentuk himpunan, alfabet *stack* dinyatakan dalam bentuk himpunan, transisi dinyatakan dalam bentuk relasi, *state* awal dinyatakan dalam bentuk anggota himpunan, dan *state* akhir dinyatakan dalam bentuk himpunan.

Sebagai contoh, Gambar 11 adalah PDA yang digunakan untuk menguji keseimbangan kurung buka dan kurung tutup.



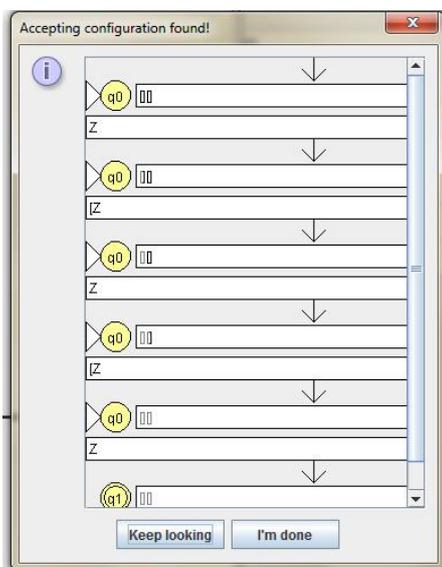
Gambar 11 Contoh PDA

Gambar 12 menunjukkan *string-string* yang dapat diterima atau ditolak. *String-string* yang dapat diterima adalah yang memiliki keseimbangan antara kurung buka dan kurung tutup, sedangkan *string-string* ditolak adalah yang memiliki keseimbangan antara kurung buka dan kurung tutup.

Input	Result
[]	Accept
][Reject
)]	Reject
[)]	Reject
[)]]	Accept
[)]]	Accept

Gambar 12 *String-string* yang Dapat Diterima atau Ditolak oleh PDA

Gambar 13 menunjukkan proses yang dilakukan oleh PDA untuk menerima suatu *string*.



Gambar 13 Proses yang Dilakukan oleh PDA untuk Menerima Suatu *String*

8. Simpulan

Dari uraian sejak awal dapat dibuat simpulan sebagai berikut :

1. Automata memiliki dasar-dasar matematika. Dasar-dasar matematikanya adalah himpunan, barisan, fungsi, dan relasi.
2. Automata dapat dinyatakan dalam bentuk gambar atau notasi matematika.
3. Perangkat lunak JFLAP dapat digunakan untuk membantu pemahaman automata secara visual.

9. Daftar Pustaka

- [1] JFLAP <http://www.cs.duke.edu/csed/jflap/>
- [2] Linz, Peter. 2001. *An Introduction to Formal Languages and Automata*, third edition, Massachusetts: Jones and Bartlett Publishers.
- [3] Lewis, Harry R. dan Papadimitriou, Christos H. 1998. *Elements of the Theory of Computation*, second edition, New Jersey: Prentice Hall Inc.
- [4] Rodger, Susan dan Finley, Thomas. 2006. *JFLAP: An Interactive Formal Languages and Automata Package*. Sudbury: Jones & Bartlett Publishers.
- [5] Sipser, Michael. 2006. *Introduction to the Theory of Computation*, second edition, Massachusetts: Thomson Course Technology.