



TEORI KOMPUTASI

PRIO HANDOKO, S.KOM., M.T.I.

Program Studi Informatika Universitas Pembangunan Jaya



BAB 3: KONSEP BAHASA DAN OTOMATA

Bab 3: Konsep Bahasa dan Otomata

Capaian Pembelajaran.

- Mahasiswa memahami teori mengenai bahasa dan otomata
- Mahasiswa memahami operasi dasar string

Agenda.

- Teori Bahasa
- Otomata
- Operasi Dasar String

Bab 3: Konsep Bahasa dan Otomata

Agenda.

- Teori Bahasa
- Otomata
- Operasi Dasar String

Teori Bahasa

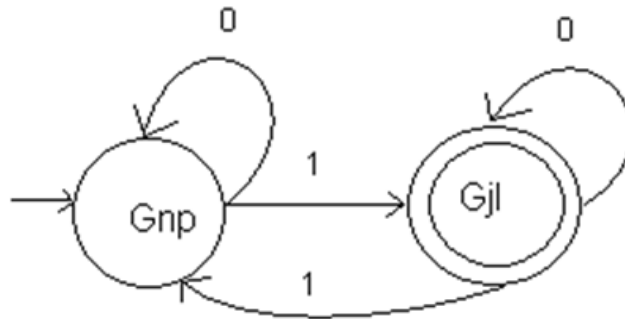
- Teori bahasa membicarakan bahasa formal (*formal language*), terutama untuk kepentingan perancangan kompilator (*compiler*) dan pemroses naskah (*text processor*).
- Bahasa formal adalah kumpulan *kalimat*. Semua kalimat dalam sebuah bahasa dibangkitkan oleh sebuah tata bahasa (*grammar*) yang sama.
- Sebuah bahasa formal bisa dibangkitkan oleh dua atau lebih tata bahasa berbeda.

Teori Bahasa

- Dikatakan bahasa formal karena *grammar* diciptakan mendahului pembangkitan setiap kalimatnya.
- Bahasa natural/manusia bersifat sebaliknya; *grammar* diciptakan untuk meresmikan kata-kata yang hidup di masyarakat.
- Dalam pembicaraan selanjutnya ‘bahasa formal’ akan disebut ‘bahasa’ saja.

Otomata

- Otomata adalah mesin abstrak yang dapat mengenali (*recognize*), menerima (*accept*), atau membangkitkan (*generate*) sebuah kalimat dalam bahasa tertentu.



Konsep Bahasa dan Otomata | 7

Otomata

Terminology.

- Simbol adalah sebuah entitas abstrak (seperti halnya pengertian *titik* dalam geometri). Sebuah huruf atau sebuah angka adalah contoh simbol.
- String adalah deretan terbatas (*finite*) simbol-simbol. Sebagai contoh, jika a , b , dan c adalah tiga buah simbol maka $abcb$ adalah sebuah string yang dibangun dari ketiga simbol tersebut.

Konsep Bahasa dan Otomata | 8

Otomata

- Jika w adalah sebuah string maka panjang string dinyatakan sebagai $|w|$ dan didefinisikan sebagai cacahan (banyaknya) simbol yang menyusun string tersebut. Sebagai contoh, jika $w = abcb$ maka $|w| = 4$.
- String hampa adalah sebuah string dengan nol buah simbol. String hampa dinyatakan dengan simbol ε (atau \wedge) sehingga $|\varepsilon| = 0$. String hampa dapat dipandang sebagai simbol hampa karena keduanya tersusun dari nol buah simbol.
- Alfabet adalah himpunan hingga (*finite set*) simbol-simbol.

Operasi Dasar String

Jika diberikan dua string : $x = abcd$, dan $y = 1234$.

- *Prefix* string w adalah string yang dihasilkan dari string w dengan menghilangkan *nol* atau lebih simbol-simbol paling belakang dari string w tersebut.
- *ProperPrefix* string w adalah string yang dihasilkan dari string w dengan menghilangkan *satu* atau lebih simbol-simbol paling belakang dari string w tersebut.

Operasi Dasar String

- *Postfix* (atau *Suffix*) string w adalah string yang dihasilkan dari string w dengan menghilangkan *nol* atau lebih simbol-simbol paling depan dari string w tersebut.
- *ProperPostfix* (atau *ProperSuffix*) string w adalah string yang dihasilkan dari string w dengan menghilangkan *satu* atau lebih simbol-simbol paling depan dari string w tersebut.
- *Head* string w adalah simbol paling depan dari string w .

Operasi Dasar String

- *Tail* string w adalah string yang dihasilkan dari string w dengan menghilangkan simbol paling depan dari string w tersebut.
- *Substring* string w adalah string yang dihasilkan dari string w dengan menghilangkan *nol* atau lebih simbol-simbol paling depan dan/atau simbol-simbol paling belakang dari string w tersebut.

Operasi Dasar String

- *ProperSubstring* string w adalah string yang dihasilkan dari string w dengan menghilangkan *satu* atau lebih simbol-simbol paling depan dan/atau simbol-simbol paling belakang dari string w tersebut.
- *Subsequence* string w adalah string yang dihasilkan dari string w dengan menghilangkan *nol* atau lebih simbol-simbol pada posisi sembarang dari string w tersebut.

Operasi Dasar String

- *ProperSubsequence* string w adalah string yang dihasilkan dari string w dengan menghilangkan *satu* atau lebih simbol-simbol pada posisi sembarang dari string w tersebut.
- *Concatenation* adalah penyambungan dua buah string. Operator concatenation adalah *concate* atau tanpa lambang apapun.
- *Alternation* adalah pilihan satu di antara dua buah string. Operator alternation adalah *alternate* atau $|$.

Operasi Dasar String

- Kleene Closure: $x^* = \varepsilon / x / xx / xxx / \dots = \varepsilon / x / x^2 / x^3 / \dots$

Kleene Closure merupakan operasi dalam logika matematika dan ilmu komputer yang hanya menggunakan sebuah *operand* (*unary*), baik pada set string /set simbol/karakter yang lebih dikenal sebagai konstruksi monoid bebas (semua elemennya adalah urutan hingga (string) dari nol atau lebih elemen dari himpunan).

- Positive Closure: $x^+ = x / xx / xxx / \dots = x / x^2 / x^3 / \dots$

Positive Closure merupakan sebuah *kleene closure* yang semua elemennya adalah urutan hingga (string) dari satu atau lebih elemen dari himpunan.

Konsep Bahasa dan Otomata | 15

Operasi Dasar String

Sifat Operasi.

- Tidak selalu berlaku: $x = \text{Prefix}(x)\text{Postfix}(x)$
- Selalu berlaku: $x = \text{Head}(x)\text{Tail}(x)$
- Tidak selalu berlaku: $\text{Prefix}(x) = \text{Postfix}(x)$ atau $\text{Prefix}(x) \neq \text{Postfix}(x)$
- Selalu berlaku: $\text{ProperPrefix}(x) \neq \text{ProperPostfix}(x)$
- Selalu berlaku: $\text{Head}(x) \neq \text{Tail}(x)$

Konsep Bahasa dan Otomata | 16

Operasi Dasar String

Sifat Operasi.

- Tidak selalu berlaku: $x = \text{Prefix}(x)\text{Postfix}(x)$

Contoh: $x = abcd$

1. $\text{prefix}(x) = ab$; $\text{postfix}(x) = cd$
maka, $\text{prefix}(x)\text{postfix}(x) = abcd = x$ ☒
2. $\text{prefix}(x) = abc$; $\text{postfix}(x) = bcd$
maka, $\text{prefix}(x)\text{postfix}(x) = abcbcd \neq x$ ☐

Operasi Dasar String

- Selalu berlaku: $x = \text{Head}(x)\text{Tail}(x)$

Contoh: $x = abcd$

$\text{head}(x) = a$; $\text{tail}(x) = bcd$;

maka, $\text{head}(x)\text{tail}(x) = abcd = x$

- Selalu berlaku: $\text{ProperPrefix}(x) \neq \text{ProperPostfix}(x)$

Contoh: $x = abcd$

$\text{properprefix}(x) = ab$; $\text{properpostfix}(x) = cd$

maka, $\text{properprefix}(x) \neq \text{properpostfix}(x)$ ☒

Operasi Dasar String

- Selalu berlaku: **Head(x) \neq Tail(x)**

Contoh: $x = abcd$

$head(x) = a$; $tail(x) = bcd$, maka $head(x) \neq tail(x)$ ☒

- Tidak selalu berlaku: **Prefix(x) = Postfix(x) atau Prefix(x) \neq Postfix(x)**

Contoh: $x = abcd$

$prefix(x) = abcd$; $postfix(x) = abcd$, maka $prefix(x) = postfix(x)$ ☒

$prefix(x) = ab$; $postfix(x) = abc$, maka $prefix(x) \neq postfix(x)$ ☐

$prefix(x) = cd$; $postfix(x) = cd$, maka $prefix(x) = postfix(x)$ ☒

OPERASI DASAR STRING

- Setiap ***Prefix(x)***, ***ProperPrefix(x)***, ***Postfix(x)***, ***ProperPostfix(x)***, ***Head(x)***, dan ***Tail(x)*** adalah ***Substring(x)***, tetapi tidak sebaliknya.

Contoh: $x = abcd$

$prefix(x) = \text{abc}$, $properprefix(x) = \text{ab}$, $postfix(x) = \text{bcd}$,

$properpostfix(x) = \text{cd}$, $head(x) = \text{a}$, dan $tail(x) = \text{bcd}$.

$Substring(x) = \text{abc}, \text{a}, \text{bcd}, \text{cd}, \text{ab}, \dots$ ☒

$Substring(x) = \text{acd}, \text{abd}, \text{bd}, \dots$ ☐

OPERASI DASAR STRING

- Setiap ***Substring(x)*** adalah ***Subsequence(x)***, tetapi tidak sebaliknya.

Contoh: $x = abcd$

$\text{substring}(x) = bc$; $\text{subsequence}(x) = bc$

maka, $\text{substring}(x) = \text{subsequence}(x)$ ✓

$\text{subsequence}(x) = ad$; $\text{substring}(x) = bc$

maka, $\text{subsequence}(x) \neq \text{substring}(x)$ ✓

Operasi Dasar String

- Operasi Dua sifat aljabar ***concatenation***:
 - ***concatenation*** bersifat asosiatif: $x(yz) = (xy)z$
 - Elemen identitas operasi ***concatenation*** adalah ε : $\varepsilon x = x\varepsilon = x$
- Tiga sifat aljabar ***alternation***:
 - Operasi ***alternation*** bersifat komutatif: $x \mid y = y \mid x$
 - Operasi ***alternation*** bersifat asosiatif: $x \mid (y \mid z) = (x \mid y) \mid z$
 - Elemen identitas operasi ***alternation*** adalah dirinya sendiri: $x \mid x = x$

OPERASI DASAR STRING

- Sifat distributif *concatenation* terhadap alternation: $x (y \mid z) = xy \mid xz$

TEORI KOMPUTASI



UNTIL NEXT WEEK...