I. PENDAHULUAN

- Komputer digunakan sebagai alat bantu untuk menyelesaikan pekerjaan(*task*). Dua pertanyaan yang seringkali muncul sehubungan dengan pekerjaan tersebut adalah :
  1. Dapatkah pekerjaan tersebut dilaksanakan oleh komputer?
  2. Bila jawabannya dapat, bagaimana cara komputer melaksanakan pekerjaan tersebut?

  Model komputasi berguna untuk menjawab kedua pertanyaan tersebut diatas.
- Terdapat tiga jenis struktur yang digunakan dalam model komputasi :
  1. Grammar (tata bahasa)

     Digunakan untuk membangkitkan kalimat dari suatu bahasa dan untuk memeriksa apakah suatu kalimat termasuk di dalam bahasa tersebut.
  2. *Finite State Machine*(FSM)

     Digunakan untuk mengenali kalimat dari suatu bahasa dan untuk menghasilkan keluaran sesuai dengan spesifikasi yang digunakan.

     Konsep tatabahasa dan FSM keduanya saling berkaitan, dan antara keduanya dapat saling ditransformasikan satu sama lain.
  3. Mesin Turing

     Merupakan pengenal bahasa yang lebih kompleks, yang dalam pengenalannya mesin ini dilengkapi dengan memori. Mesin turing dapat mengenali kalimat yang tidak dapat dikenali oleh FSM sekalipun. Mesin turing adalah model komputasi yang paling umum serta dapat melakukan pekerjaan apa saja yang dapat dilaksanakan oleh komputer.

I.1. Bahasa Alami dan Bahasa Formal

- Menurut kamus *Webster*, bahasa adalah ”*the body of words and methods of combining words used and understood by a considerable community*”.
- Sedangkan menurut Kamus Besar Bahasa Indonesia(KBBI), bahasa adalah :
  1. Sistem lambang bunyi berartikulasi (yang dihasilkan oleh alat-alat ucap) yang bersifat sewenang-wenang dan konvensional yang dipakai sebagai alat komunikasi untuk melahirkan perasaan dan pikiran.
  2. Perkataan-perkataan yang dipakai oleh suatu bangsa.
- Menurut kedua definisi diatas, ungkapan dari suatu bahasa adalah kalimat atau perkataan. Kalimat dalam sebuah bahasa dibentuk dengan menggabungkan satu atau lebih kata-kata.
- Dalam pembentukan kalimat, perhatian tertuju pada sintaks dan bentuk kalimat, bukan pada semantik atau makna kalimatnya. Sebagai contoh :

  Elang menyambar ayam

  Adik makan dengan lahap

  Adalah kalimat yang sah sesuai dengan kaidah tata bahasa dari Bahasa Indonesia. Contoh kalimat berikut :

  Rudi menggigit anjing

  Juga benar secara sintaks meskipun maknanya secara sense tidak ada. Sedangkan kalimat berikut :

  dengan cepat ayah matematika menutup

  bukan kalimat yang sah secara sintaks karena tidak mengikuti kaidah tata bahasa.
- Bahasa Alami adalah bahasa komunikasi yang digunakan oleh manusia, yaitu bahasa ucap seperti Bahasa Indonesia, Inggris, dll. Sintaks bahasa alami sangat rumit dan kita tidak mungkin menspesifikasikan semua aturan sintaksnya.
- Bahasa Formal adalah bahasa yang kaidah sintaksnya dapat dispesifikasikan secara matematis dengan baik. Kaidah bahasa dalam teori bahasa formal tidak hanya bermanfaat untuk studi linguistik bahasa alami

seperti penterjemahan secara otomatis dari suatu bahasa ke bahasa lain, tetapi juga berguna untuk studi bahasa pemrograman.

I.2. Alfabet, *String*, dan Bahasa

- Di dalam teori bahasa formal, bahasa didefinisikan secara matematis. Sebelum membahas lebih lanjut mengenai definisi bahasa, kita harus mengerti beberapa terminologi(istilah) yang digunakan disini.
- **Alfabet** (adalah *vocabulary*) adalah himpunan terbatas simbol.

  Contoh :

  a. Alfabet Latin {A, B, C, .., Z)
  b. Alfabet Yunani {α,β,γ,..,ω}
  c. Alfabet Biner, {0,1}

- **String** adalah barisan(*sequence*) yang disusun oleh simbol-simbol alfabet. Sebuah *string* dengan panjang n (n >= 1) yang dibentuk dari alfabet A disusun oleh barisan n simbol :

  $$\mathbf{a_1 a_2 a_{3,..} a_n,} \qquad \mathbf{a_I \in A}$$

  istilah lain untuk *string* adalah kalimat atau *word*.

  - Panjang *string x* dituliskan sebagai IxI. *String* kosong (*null string* atau *empty sentence* (ε)) adalah barisan simbol dengan panjang 0 dan tidak berisi simbol apapun.
  - Jika A adalah alfabet, maka A^n menyatakan himpunan semua *string* dengan panjang n yang dibentuk dari himpunan A.
  - A* adalah himpunan semua rangkaian simbol dari himpunan A yang terdiri dari 0 simbol (*string* kosong), satu simbol, dua simbol, tiga simbol,.., atau dinyatakan sebagai :

    $$\mathbf{A^* = A^0 \cup A^1 \cup A^2 \cup ...}$$

    Himpunan A* disebut ***transitive closure* atau *Kleen Closure*.**
  - A^+ (*positive closure*) untuk menyatakan himpunan V* - ∈. Jadi, A^+ diperoleh dari A* dengan mengeluarkan string kosong, atau :

    $$\mathbf{A^+ = A^1 \cup A^2 \cup A^3 \cup ...}$$

- Contoh :

  Misalkan A={0,1} adalah sebuah alphabet yang disusun dari simbol 0 dan 1. Maka A* :

  A^0={ ε }

  A^1={ 0,1 }

  A^2={ 00,01,10,11 }, dan seterusnya

  Apabila A^+ :

  A^1={ 0,1 }

  A^2={ 00,01,10,11 }, dan seterusnya

- **Definisi bahasa(*language*)**

  Jika A adalah alfabet, maka bahasa (pada alfabet A) adalah bagian dari dari himpunan A*. Sebagai contoh, misalkan {a,b,c}, maka himpunan-himpunan berikut ini merupakan bahasa pada alfabet A :

  L_1 = {a, aa, bc, ac, abc, cab}

  L_2 = {aba, aabaa}

  L_3 = { }

  L_4 = {a^icb^i I i>=1}

  (Keterangan : pada L_4 a^i berarti barisan *i* buah *a*. Jadi a^icb^i berarti kalimat yang disusun oleh *i* buah *a* diikuti dengan *c*, diikuti dengan *i* buah *b*)

I.3 Tata Bahasa

- Jika diberikan spesifikasi suatu bahasa, ada dua masalah utama yang berkaitan dengan masalah tersebut :
  1. Bagaimana membangkitkan satu atau lebih kalimat didalam bahasa tersebut ?
     - Masalah pembangkitan (*Generation*)
  2. Bagaimana menentukan apakah suatu kalimat termasuk di dalam bahasa tersebut atau tidak?
     - Masalah Pengenalan (*Recognition*)

- Aturan yang digunakan untuk membangkitkan atau mengenali kalimat di dalam suatu bahasa dinamakan tata bahasa(*grammar*).
- **Contoh :** Tata bahasa dibawah ini adalah sebagian dari tata bahasa untuk membangkitkan atau mengenali kalimat di dalam Bahasa Inggris :
  1. Sebuah **sentence** dibentuk dari **noun phrase** diikuti dengan **verb phrase.**
  2. Sebuah **sentence** dibentuk dari **noun phrase** diikuti dengan **verb phrase** dan **noun phrase.**
  3. dll
- Kita dapat menuliskan semua kaidah tata bahasa di atas secara ringkas sebagai berikut :

  <sentence> → <noun phrase> <verb phrase>

  <sentence> → <noun phrase> <verb phrase> <noun phrase>

  <noun phrase> → <adjective><noun phrase>

  <noun phrase> → <adjective><singular noun >

  <verb phrase > → <singular verb> <adverb>

  <verb phrase > → <singular verb>

  <adjective> → *the*

  <adjective> → *a*

  <adjective> → *little*

  <singular noun> → *boy*

  <singular noun> → *dog*

  <singular verb> → *runs*

  <singular verb> → *bites*

  <adverb> → *quickly*
- Tanda panah (→) pada kaidah di atas menyatakan bahwa item di sebelah kiri panah dapat diganti dengan item di sebelah kanan panah.
- Proses pembangkitan kalimat *the little boy runs quickly* dapat diturunkan sebagai berikut :

  <sentence> → <noun phrase> <verb phrase>

  → <adjective> <noun phrase> <verb phrase>

→ *the* <noun phrase> <verb phrase>

→ *the* <adjective> <singular noun > <verb phrase>

→ *the little* <singular noun > <verb phrase>

→ *the little boy* <verb phrase>

→ *the little boy* <singular verb> <adverb>

→ *the little boy runs* <adverb>

→ *the little boy runs quickly*

- Proses pembangkitan kalimat diatas juga dapat digambarkan dalam suatu diagram pohon- yang disebut *pohon parsing*.
- Unsur-unsur tata bahasa :

  Secara formal, suatu tata bahasa terdiri dari empat unsur :

  1. Himpunan berhingga **terminal T**
  2. Himpunan berhingga **non –terminal N**
  3. Himpunan berhingga **aturan produksi P**
  4. **Simbol awal S € N**

- Penjelasan :

  1. Simbol terminal didalam *T* adalah symbol yang dapat digunakan untuk membuat kalimat/string di dalam bahasa itu. Simbol terminal biasanya dilambangkan dengan huruf kecil, seperti a,b,c,… atau angka 0,1,..

     Contoh : *T* = {a,b,c}
  2. Simbol non terminal di dalam *N* adalah lambang antara yang digunakan untuk mendeskripsikan struktur kalimat. Simbol non-terminal biasanya dilambangkan dengan hruf besar, seperti A,B,C,..

     Contoh : *N* = {S,A,B}
  3. Produksi adalah kaidah/aturan tata bahasa yang mengatur cara pembentukan kalimat di dalam bahasa.

     Setiap aturan dituliskan dalam bentuk :

     $$\alpha \rightarrow \beta$$

     yang dalam hal ini, **α dan β** adalah string yang dibentuk dari himpunan *T U N*, dan **α** harus berisi paling sedikit satu simbol non-terminal.

4. Simbol awal adalah simbol non-terminal khusus yang mengawali pembangkitan kalimat di dalam bahasa tersebut.

- Keempat unsur tata bahasa tersebut sering dilambangkan dengan :

$$\boldsymbol{G = (T,N,P,S)}$$

Bahasa yang dihasilkan oleh *G* ditulis sebagai L(*G*), yaitu himpunan *string* yang dapat diturunkan dan simbol awal *S* dengan menerapkan aturan produksi yang terdapat di dalam *P*.

- Contoh :

Tinjau tata bahasa $\boldsymbol{G = (T,N,P,S)}$, dengan $\boldsymbol{T=\{a,b\},\ N=\{S,A\},\ P=\{S\rightarrow aA, S\rightarrow b,\ A\rightarrow aa\}}$

Apakah *L(G)* dari Tata bahasa *G* ?

Dengan mencoba semua aturan produksi dari simbol awal *S* sebagai berikut :

(i) $\boldsymbol{S \rightarrow aA}$

$\boldsymbol{\rightarrow aaa}$

(i) $\boldsymbol{S \rightarrow b}$

maka hanya terdapat dua *string* yang dapat dibangkitkan, yaitu *b* dan *aaa*, tidak ada yang lain. Jadi *L(G) = {b, aaa}*.

1.4 Kelas Tata Bahasa dan kelasa Bahasa

- Berdasarkan bentuk aturan produksi $\alpha \rightarrow \beta$, tata bahasa formal dibagi ke dalam empat kelas (jenis). Keempat kelas itu disebut juga dengan *Hirarki Chomsky*, sebagaimana diklasifikasikan oleh Noam Chomsky.
- Kelas bahasa ditentukan oleh kelas bahasanya. Jadi, bahasa kelas *i* adalah bahasa yang dapat dispesifikasi oleh tata bahasa kelas *i*.
- Di dalam pembahasan berikut, kita menggunakan A dan B untuk menyatakan non-terminal, a dan b untuk menyatakan terminal, dan α dan β untuk menyatakan string yang dibentuk dari simbol terminal dan simbol non terminal.
- Keempat kelas tata bahasa tersebut adalah sebagai berikut :

1. Kelas tata bahasa regular (regular grammar)

- Merupakan kelas tata bahasa jenis-3. Kelas ini dicirikan jika semua aturan produksinya berbentuk :

  $$\alpha \rightarrow \beta$$

  dimana : $\alpha \in N$, $\beta \in T \cup N$ dengan ketentuan sebagai berupa terminal atau sebuah terminal diikuti dengan sebuah non-terminal. Contoh : A → a ; A → aA
- Bahasa yang dispesifikasi oleh tata bahasa ini disebut **bahasa regular** (regular language)
- Mesin yang mengenal **bahasa regular** adalah ***Finite State Automata.***

2. Kelas tata bahasa bebas konteks (context free grammar)

- Merupakan kelas tata bahasa jenis-2. Kelas ini dicirikan jika semua aturan produksinya berbentuk :

  $$\alpha \rightarrow \beta$$

  dimana :

  $\alpha \in N$, dengan ketentuan sebuah non-terminal

  $\beta \in T \cup N$.

  $|\alpha| <= |\beta|$
- Contoh : A → aBB ; A → aA ; B→bA
- Mesin yang mengenal **bahasa bebas konteks** adalah ***Push Down Automata (PDA).***

3. Kelas tata bahasa peka-konteks (context sensitive grammar)

- Merupakan kelas tata bahasa jenis-1. Kelas ini dicirikan jika semua aturan produksinya berbentuk :

  $$\alpha \rightarrow \beta$$

  dimana :

  $\alpha \in T \cup N$

  $\beta \in T \cup N$

  $|\alpha| <= |\beta|$
- Contoh : S → aSBC ; S → aBC ; CB→BC ; bB→bb ; bC→bc; cC→cc

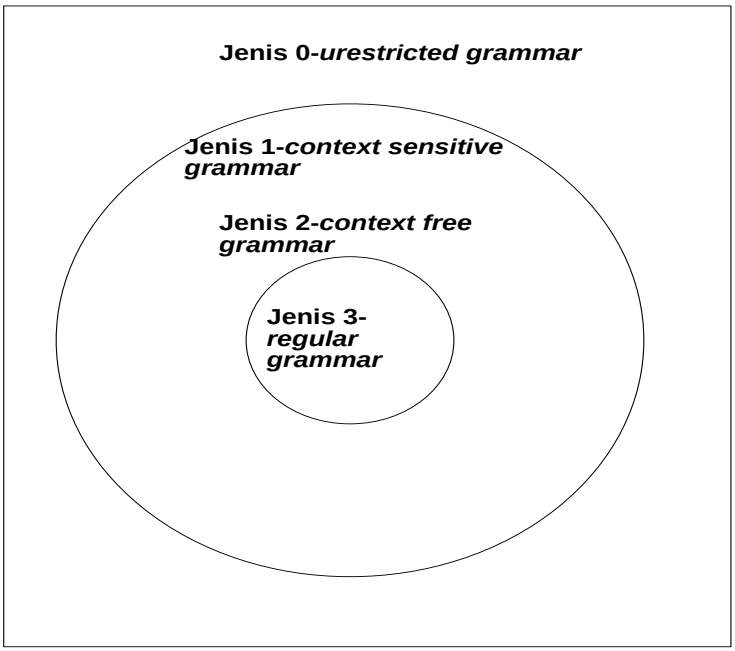
- Mesin yang mengenal **bahasa peka-konteks** adalah ***Linear Bounded Automata (LBA).***

4. Kelas tata bahasa tanpa-pembatasan (*urestricted grammar*)

Merupakan kelas tata bahasa jenis-0. Kelas ini tidak memiliki pembatasan untuk aturan produksinya

- Contoh : aS → aSBC ; S → aBC ; CbB→BC ; bB→bb ; bC→bc; cC→ccB
- Mesin yang mengenal **bahasa tanpa-pembatasan** adalah **Mesin Turing (*Turing Machine*).**

Hubungan setiap bahasa digambarkan dalam diagram venn berikut :



LATIHAN

1. Pilihlah dari sekian aturan produksi dibawah ini kedalam level-level bahasa yang ada:
   a. E → fgH
   b. E → HgB
   c. EF → a
   d. A→ aeGab
   e. Abc → dEF
   f. D → Ddd
   g. A → AAA
   h. FGH → abc
2. Buatlah alasan-alasan untuk setiap jawaban dari soal diatas !