

## APLIKASI PEMERIKSAAN TATA BAHASA DALAM KALIMAT BAHASA INDONESIA MENGGUNAKAN ALGORITMA *LEFT CORNER PARSING*

Olvina Rante Allun

Debby Paseru\*

Immanuela Saputro

Program Studi Teknik Informatika  
Fakultas Teknik – Univ. De La Salle  
Kombos Kairagi I Manado

[olvinarante@gmail.com](mailto:olvinarante@gmail.com)

[dpaseru@unikadelasalle.ac.id](mailto:dpaseru@unikadelasalle.ac.id)

[isaputro@unikadelasalle.ac.id](mailto:isaputro@unikadelasalle.ac.id)

### ABSTRAK

Sebagai alat komunikasi penduduk Indonesia dan alat pemersatu penduduk antar pulau di seluruh wilayah Indonesia, Bahasa Indonesia perlu dipahami dengan benar, baik dari cara pengucapannya maupun cara penulisannya. Dalam penulisan kata atau kalimat dalam Bahasa Indonesia perlu memperhatikan dan mengikuti aturan penulisan bahasa yang baik dan benar. Untuk mengetahui penulisan bahasa Indonesia khususnya tata bahasa apakah sesuai dengan aturan yang benar atau tidak, dapat dilakukan dengan melihat pola kalimat bahasa Indonesia tersebut.

*Left corner parsing* adalah salah satu metode *parsing* yang menggabungkan *Top Down* dan *Bottom Up parsing*, yaitu dengan mengambil prediksi dari *Top Down* dan *Bottom Up*. Dari kedua metode tersebut kalimat akan diuraikan menjadi kata per kata dan dicari aturan bahasa Indonesia yang telah ditentukan sebelumnya yang sesuai dengan kata yang sedang diuraikan tersebut. Bila ditemukan suatu aturan yang sesuai dengan pola kalimat, maka proses uraian (*parsing*) tersebut telah berhasil. Penelitian ini membahas dan membuat sebuah alat bantu yang berfungsi untuk mengenali struktur yang terdapat dalam kalimat bahasa Indonesia yang dimasukkan apakah sesuai dengan tata bahasa Indonesia atau tidak.

Penelitian ini dibuat dengan menggunakan metodologi RUP (*Rational Unified Process*), bahasa pemrograman *Visual Basic 6.0* dengan kaskas yang digunakan UML (*Unified Modelling Language*).

Hasil dari pengujian yang dilakukan menyimpulkan bahwa aplikasi yang dibuat pada penelitian ini dapat menghasilkan bentuk kalimat yang sesuai dengan tata bahasa Indonesia yang berlaku.

**Kata kunci:** *Parsing*, *Left Corner*, Bahasa Indonesia

### I. Pendahuluan

Sebagaimana disebutkan dalam Undang-Undang Dasar RI 1945, Pasal 36, bahwa “Bahasa Negara adalah Bahasa Indonesia adalah juga bahasa persatuan bangsa Indonesia”. Meskipun demikian, hanya sebagian kecil dari penduduk Indonesia yang benar-benar menggunakannya sebagai bahasa sehari-hari karena masih banyak yang menggunakan bahasa daerah masing-masing sebagai bahasa ibu, seperti bahasa Manado, bahasa Jawa, bahasa Sunda, dan lain sebagainya. Akibatnya bahasa Indonesia menjadi bahasa sehari-hari kedua dalam kehidupan penduduk dan bahasa Indonesia hanya dipakai untuk situasi tertentu. Dalam perbendaharaan kosa kata bahasa Indonesia kata-kata yang berlatar belakang kata-kata dari bahasa Melayu paling menonjol.

Dalam tahapan pendidikan formal sejak tingkat pendidikan dasar, setiap orang telah menerima teori berbahasa Indonesia yang benar termasuk dalam hal tata bahasa namun dalam prakteknya, seringkali ditemukan pola-pola kalimat yang tidak sesuai dengan kaidah tata bahasa Indonesia yang benar. Untuk dapat berkomunikasi dengan baik dan dapat dipahami oleh orang lain, maka tata bahasa yang dipakai harus pula sesuai dengan kaidah berbahasa Indonesia yang benar. Tak jarang dalam berkomunikasi seseorang memakai bentuk kalimat tertentu tanpa menghiraukan apakah kalimat yang digunakan itu sesuai dengan tata bahasa Indonesia yang berlaku atau tidak.

Olehnya untuk memeriksa kebenaran tata bahasa Indonesia yang digunakan maka akan dibangun sebuah aplikasi yang dapat memeriksa kalimat-kalimat bahasa Indonesia dalam naskah tertentu untuk menentukan bagian-bagian mana yang sesuai dengan tata bahasa Indonesia yang benar. Aplikasi ini menggunakan algoritma *Left Corner Parsing* yang memiliki metode analisis kemungkinan yang didapat dan kemudian pencarian langsung untuk menghindari masalah umum dalam kata yang sering bukan merupakan akhir dari suatu kalimat. Dengan memakai algoritma *Left Corner Parsing* ini akan lebih mudah dihasilkan bentuk kalimat yang lebih tepat dan mendekati pola kalimat yang sebenarnya sesuai dengan tata bahasa Indonesia yang berlaku.

#### I.1. Tinjauan Pustaka yang Berkaitan

Terdapat beberapa penelitian yang mirip dengan penelitian yang dilakukan, namun hanya dua penelitian yang akan dibahas, yaitu Pemeriksaan Tata Bahasa dalam Kalimat Bahasa Inggris Menggunakan Algoritma Left

Corner Parsing oleh Maman Abdurohman, Syamsul Hadi dan Dede Rohidin (2006) dan Implementasi Left Corner Parser pada Perancangan Aplikasi Pemeriksaan Tata Bahasa dalam Kalimat Bahasa Indonesia oleh Vita Meriati Pandiangan (2015). Kedua penelitian menggunakan algoritma yang sama namun penelitian oleh Maman dkk (2006) diterapkan pada Kalimat Bahasa Inggris. Bahasa pemrograman yang digunakan juga sama namun memiliki versi yang berbeda, di mana penelitian yang kami lakukan juga akan menggunakan *Visual Basic 6.0* yang sama dengan penelitian oleh Maman dkk (2006), sedangkan penelitian oleh Pandiangan (2015) menggunakan *Visual Studio 8.0*. Dalam melakukan pemeriksanaan tata bahasa, penelitian pertama menggunakan pola Subjek Predikat Objek (SPO) untuk kalimat bahasa Inggris, penelitian kedua menggunakan pola SPOK (SPO Keterangan) dan untuk penelitian yang akan dilakukan akan menggunakan beberapa pola seperti SPOK, SPO, SPK dan SP. Bentuk pola kalimat yang akan digunakan terbatas pada bentuk pola kalimat positif dan bentuk pola kalimat negatif, tidak termasuk kalimat pertanyaan dan tidak memiliki anak kalimat. Selain itu pula kalimat yang diperiksa hanya berdasarkan pada kata yang ada di dalam kamus kata.

### 1.2. *Objektivitas dari Makalah*

Tujuan dari penelitian ini adalah untuk memeriksa tata bahasa dalam suatu kalimat melalui aplikasi yang dapat melakukan *parsing* terhadap suatu kalimat menggunakan algoritma *Left Corner Parsing*.

## II. Landasan Teori

Demi memperjelas penelitian yang dilakukan maka perlu disajikan teori-teori pendukung sebagai berikut:

### II.1. *Pola Dasar Kalimat Bahasa Indonesia*

Kalimat yang digunakan biasanya dapat dikembalikan ke sejumlah kalimat dasar yang sangat terbatas atau dengan kata lain, semua kalimat yang digunakan berasal dari beberapa pola kalimat dasar saja. Sesuai dengan kebutuhan, kalimat dasar tersebut dapat dikembangkan di mana pengembangannya itu harus didasarkan pada kaidah yang berlaku.

Kalimat dasar adalah kalimat yang berisi informasi pokok dalam struktur inti sebelum mengalami perubahan. Perubahan itu dapat berupa penambahan unsur seperti penambahan keterangan kalimat ataupun keterangan subjek, predikat, objek, ataupun keterangan. Kalimat dasar dapat dibedakan ke dalam empat tipe sebagai berikut (Mulyono 2012):

#### 1. Kalimat Dasar berpola Subjek dan Predikat

Kalimat dasar tipe ini memiliki unsur subjek dan predikat. Predikat kalimat untuk tipe ini dapat berupa kata kerja, kata benda, kata sifat, atau kata bilangan. Misalnya:

- |                  |                                  |                   |                            |
|------------------|----------------------------------|-------------------|----------------------------|
| a. Mereka /<br>S | sedang bekerja<br>P (kata kerja) | b. Ayahnya /<br>S | guru SMA<br>P (kata benda) |
|------------------|----------------------------------|-------------------|----------------------------|

#### 2. Kalimat Dasar berpola Subjek, Predikat dan Objek

Kalimat dasar tipe ini memiliki unsur subjek, predikat, dan objek. Subjek berupa nomina atau frasa nominal, predikat berupa verba transitif, dan objek berupa nomina atau frasa nominal. Misalnya:

- |                  |                        |                      |
|------------------|------------------------|----------------------|
| a. Mereka /<br>S | sedang menyusun /<br>P | karangan ilmiah<br>O |
|------------------|------------------------|----------------------|

#### 3. Kalimat Dasar berpola Subjek, Predikat, dan Keterangan

Kalimat dasar tipe ini memiliki unsur subjek, predikat dan harus memiliki unsur keterangan karena diperlukan oleh predikat. Subjek berupa nomina atau frasa nominal, predikat berupa verba intrasitif, dan keterangan berupa frasa berpreposisi. Misalnya:

- |               |                |                  |
|---------------|----------------|------------------|
| Mereka /<br>S | berasal /<br>P | dari Manado<br>K |
|---------------|----------------|------------------|

#### 4. Kalimat Dasar berpola Subjek, Predikat, Objek, dan Keterangan

Kalimat dasar ini memiliki unsur subjek, predikat, objek, dan keterangan. Subjek berupa nomina atau frasa nomina, predikat berupa verba intrasitif, objek berupa nomina atau frasa nominal, dan keterangan berupa frasa berpreposisi. Misalnya: Kami / memasukkan / pakaian / ke dalam lemari

- |   |   |   |   |
|---|---|---|---|
| S | P | O | K |
|---|---|---|---|

### II.2. *Parsing*

*Parsing* adalah proses pengenalan dan pengambilan token hasil dari sekumpulan unit dokumen, yang biasanya kata-kata. Proses *parsing* tidak hanya dapat dilakukan dalam proses *information retrieval* (temu kembali informasi), melainkan juga pada bidang lain seperti pada pembuatan sebuah *compiler* dan bahasa alami. *Parser* yaitu program yang melakukan proses *parsing*. Untuk pemrosesan, dokumen dipilah menjadi unit-unit yang lebih kecil misalnya berupa kata, frasa atau kalimat. Unit pemrosesan tersebut disebut sebagai token. Oleh karena itu

bagian dasar dalam *parsing* adalah algoritma pengambil token dari teks yang disebut *tokenizer*. Proses ini memerlukan pengetahuan bahasa untuk menangani karakter-karakter khusus, dan menentukan batasan satuan unit dalam dokumen (Rodriguez 2010).

*Parsing* juga adalah proses menentukan apakah sekumpulan *string* yang dibuat oleh sebuah *grammar*. *Parser* ini diharapkan dapat melaporkan kesalahan-kesalahan sintaks yang ada dalam bahasa tersebut. *Parser* mempunyai peranan dalam sebuah pengenalan sintaksis sebagai pengolah data dari pengenalan leksikal dan membuat pohon *parser* sehingga dapat dihasilkan informasi keluaran berupa pola kalimat yang telah dikenali.

### II.3. Algoritma Left Corner Parsing

Ada dua jenis algoritma *parsing* yang biasa digunakan untuk sebuah tata bahasa, yaitu algoritma *Early* dan *Left Corner Parsing*. *Left Corner Parsing* merupakan gabungan antara *top down* dan *bottom up parsing* dengan mengambil prediksi dari *top down* dan penelusuran dari *bottom up*, sedangkan algoritma *Early* merupakan strategi yang hanya dilakukan secara *top down* saja. *Left Corner Parsing* merupakan strategi *parsing* yang menggunakan data secara *bottom up* (dari aturan *left corner*) dan prediksi *top down* (dari aturan sisanya) (Rodriguez 2010).

Olehnya metode yang dapat menggabungkan *top down* dan *bottom up parsing* (*Left Corner Parsing*) adalah metode yang cocok untuk pemeriksaan kalimat karena metode ini dapat membaca masukan per kata dan memprosesnya dengan aturan *left corner*. Metode ini juga melakukan pemeriksaan kebenaran pola untuk setiap kalimat yang diterima dengan mencocokkan pola kalimat yang ada setelah didapatkan pola kalimat tersebut dengan aturan *left corner*. Secara teoritis metode ini akan efisien jika digunakan pada kalimat yang tidak memiliki pola kalimat yang bercabang pada suatu kalimat yang sedang dicari pola kalimatnya.

Dalam penggunaan *Left Corner Parsing* akan digunakan *context free grammar* sebagai bahasa yang akan diolah oleh *parser*. Untuk *Left Corner Parsing* ini akan digunakan beberapa istilah dalam mengolah masukan sehingga didapatkan hasil pola kalimat (Rodriguez 2010):

1. *Grammar* atau aturan tata bahasa yang berisi pola kalimat bahasa Indonesia yang berbentuk *context free grammar*. Pada *grammar* terdapat aturan produksi yang akan diproses untuk dicari *left corner*.

Contoh aturan produksi:

$S \rightarrow NP.VP$ .....(1)

$NP \rightarrow Det.N$ .....(2)

Dengan  $NP \rightarrow$ frasa kata benda

$VP \rightarrow$ frasa kata kerja

2. *Stack* atau tumpukan yang berfungsi sebagai tempat penyimpanan aturan produksi yang sedang dan akan diproses.
3. *Right hand side* (RHS) adalah sisi bagian kanan dari aturan produksi, maka *right hand side* dari contoh aturan produksi di atas adalah (1) NP.VP dan det.N.
4. *Left hand side* (LHS) adalah sisi bagian kiri dari aturan produksi, maka *left hand side* dari contoh aturan produksi di atas adalah (1) S dan (2) NP.  
*Left corner* item adalah item paling kiri yang berada pada sisi bagian kanan dari sebuah aturan produksi, maka *left corner* item dari contoh aturan produksi di atas (1) NP dan (2) Det.

Cara kerja *Left Corner Parsing* ialah dengan mula-mula menerima sebuah kata, menentukan jenis kata apa yang dimulai dengan jenis kata tersebut, kemudian melakukan proses *parsing* terhadap sisa dari kata tersebut secara *top-down*. Dengan demikian proses *parsing* dimulai secara *bottom-up* dan diakhiri secara *top-down*. Algoritma *Left Corner Parsing* dapat menghasilkan bentuk tata bahasa yang digunakan oleh kalimat tersebut untuk mengetahui kesalahan tata bahasa dari kalimat tersebut. Menurut Blackburn and Striegnitz (2009), cara kerja algoritma ini adalah:

1. Ide/gagasan kunci dari uraian sudut kiri adalah untuk menggabungkan pengolahan secara *top-down* dengan proses *bottom-up* untuk menghindari terjadinya kesalahan dalam cara-cara untuk mengambil langkah yang murni *top-down* dan teknik yang murni *bottom-up*.
2. Sebelum melihat bagaimana hal ini dilakukan, harus tahu apa itu aturan sudut kiri.  
"Aturan sudut kiri" adalah simbol pertama di sisi kanan.

Misalnya, NP adalah aturan sudut kiri  $S \rightarrow NP.VP$  ,dan

IV adalah aturan sudut kiri  $VP \rightarrow IV$ .

- a) Demikianlah, dapat dikatakan bahwa Ika (dalam contoh kalimat Ika tertawa) adalah aturan leksikal sudut kiri  
 $PN \rightarrow Ika$

b) Sebuah *parser*/alat urai bergantian sudut kiri langkah pengolahan *bottom-up* dengan prediksi *top-down*.

c) Langkah-langkah pengolahan *bottom-up* bekerja sebagai berikut:

Dengan asumsi bahwa *parser* baru saja menyatakan frase kata benda, maka akan ditampilkan langkah berikutnya untuk aturan yang memiliki NP sebagai sudut kirinya.

Katakanlah menentukan  $S \rightarrow NP.VP$ .

Untuk dapat menggunakan aturan ini, harus mengenali VP sebagai hal berikutnya dalam *string* masukan.

Hal ini menyebabkan kendala *top-down*, yakni bahwa apa yang berikut dalam *string* masukan harus menjadi frase verba.

*Parser* sudut kiri akan terus bolak-balik langkah *bottom-up* (seperti yang dijelaskan di atas) dan langkah *top-down* sampai berhasil mengenali frase verba, sehingga melengkapi kalimat.

3. Sebuah *parser*/alat urai sudut kiri dimulai dengan prediksi *top-down* memperbaiki kategori yang harus diakui, seperti misalnya S.

Selanjutnya, dibutuhkan langkah *bottom-up* dan kemudian bergantian *bottom-up* dan *top-down* langkah sampai telah mencapai S.

### III. Metode Penelitian

Dalam melakukan penelitian ini ada beberapa langkah yang dilakukan yaitu:

- a. Melakukan Studi Literatur, di mana penulis mempelajari teori dasar tentang pola kalimat dalam bahasa Indonesia dan algoritma *Left Corner Parsing*. Juga melakukan pencarian literatur mengenai penelitian sejenis dari jurnal *online* dan buku. Studi ini bersifat teoritis yang menjadi dasar dalam pembuatan aplikasi.
- b. Melakukan Analisis Awal  
Penulis melakukan analisis terhadap penelitian-penelitian sebelumnya, juga menganalisis kebutuhan fungsional dari penelitian yang akan dilakukan. Hasil analisis awal terhadap penelitian-penelitian sebelumnya digunakan untuk merancang dan membuat aplikasi yang akan dijadikan alat tolak ukur pemeriksaan tata bahasa Indonesia.
- c. Pengembangan Aplikasi  
Aplikasi pemeriksa tata bahasa dibuat menggunakan *Rational Unified Process* sebagai metodologi pengembangan sistem. Aplikasi ini memiliki fungsi untuk memeriksa kalimat yang ada apakah sesuai dengan tata bahasa Indonesia yang benar atau tidak dengan menggunakan algoritma *Left Corner Parsing* dalam penelurusan kalimat.
- d. Analisis Hasil  
Aplikasi yang dibuat nantinya akan diujicobakan dengan beberapa contoh kalimat sehingga sesuai dengan pola kalimat yang ada. Selanjutnya aplikasi yang telah diujicobakan tersebut akan dievaluasi dan diperbaiki demi kebutuhan penelitian.
- e. Kesimpulan  
Tahap terakhir setelah pembuatan dokumentasi dan sistem adalah penarikan kesimpulan dari yang telah dilakukan penulis dan pemberian saran untuk pengembangan dan perbaikan penelitian selanjutnya.

### IV. Hasil Penelitian dan Pembahasan

Dalam pembuatan aplikasi pemeriksa tata bahasa diperlukan suatu analisis awal mengenai kebutuhan aplikasi yang akan dibuat dan perancangan yang tepat sesuai dengan analisis sehingga aplikasi yang dibuat menjadi baik dan terarah. Namun sebelum dijabarkan mengenai hasil analisis awal terhadap aplikasi yang akan dibuat perlu dijelaskan mengenai penerapan algoritma *Left Corner Parsing* ke sebuah kalimat.

#### IV.1. Penerapan Algoritma Left Corner Parsing

Untuk menggambarkan bagaimana kerja “*parser* (pengurai) sudut kiri”, lihat suatu contoh: Ika Tertawa.

Asumsikan bahwa memiliki tata bahasa sebagai berikut:

S → NP.VP

NP → PN

VP → IV

PN → Ika

IV → Tertawa

Keterangan: S (Kalimat), NP (Kata benda), VP (Kata Kerja), PN (Kata ganti), dan IV (Kata sifat).

Sekarang, lihat bagaimana pengenal kiri-sudut akan melanjutkan untuk mengenali *Ika tertawa*.

- 1) *Input*: “*Ika tertawa*.” – Mengenali S. ( prediksi *top-down*)

S

Ika                      tertawa

- 2) Kategori dari kata pertama dari *input* PN ( langkah *bottom-up* menggunakan aturan leksikal)

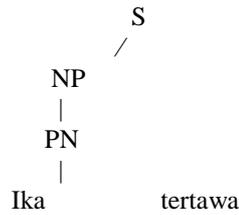
S

PN

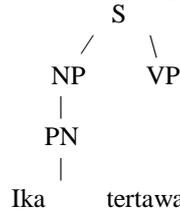
|

Ika                      tertawa

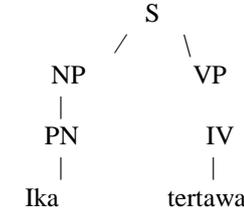
- 3) Pilih aturan yang telah PN di pojok kiri: NP → PN ( langkah *bottom-up* menggunakan aturan struktur frase)



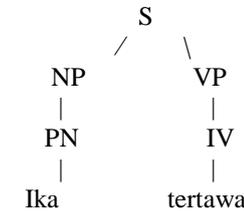
- 4) Pilih aturan yang telah NP di pojok kiri: S → NP.VP langkah *bottom-up*  
 5) Cocokkan! Sisi kiri dari aturan yang cocok dengan S, kategori berusaha untuk mengenali.



- 6) *Input*: Tertawa. Mengenali VP. ( Prediksi *top-down*)  
 7) Kategori dari kata pertama dari *input* IV. ( langkah *bottom-up*)



- 8) Pilih aturan yang telah IV di pojok kiri: VP → IV. ( langkah *bottom-up*)  
 9) Cocokkan! Sisi kiri dari aturan yang cocok dengan VP. Kategori berusaha untuk mengenali.



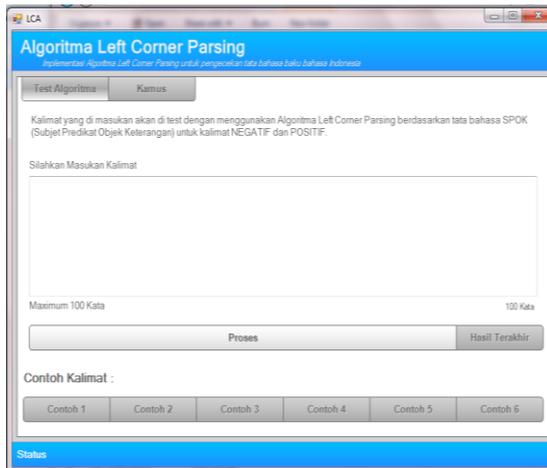
Dapat dilihat bagaimana langkah-langkah dari *bottom-up* mengaplikasikan aturan alternatif dengan prediksi *top-down* dalam contoh ini. Jadi, dari contoh kalimat “Ika tertawa” mendapatkan hasil pola kalimat yang sesuai dengan tata bahasa Indonesia yaitu memiliki dasar pola kalimat bahasa Indonesia yaitu subjek dan predikat.

#### IV.2. Pembuatan Aplikasi

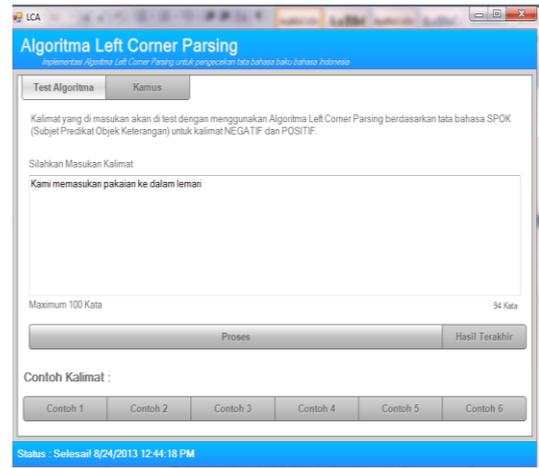
Dari analisis awal yang dilakukan diperoleh beberapa kebutuhan aplikasi pemeriksa tata bahasa sebagai berikut:

1. Aplikasi wajib memiliki *input* data berupa kalimat atau beberapa kalimat dalam bahasa Indonesia.
2. Aplikasi mampu melakukan proses *parsing* terhadap kalimat yang dimasukkan.
3. Aplikasi mampu menampilkan hasil *parsing* dari kalimat yang telah dimasukkan sesuai dengan aturan tata bahasa.
4. Aplikasi mampu mengubah kata dan jenis kata yang terdapat pada kamus kata.
5. Aplikasi mampu menambah kata dan jenis kata baru supaya jumlah kata dan kalimat yang dapat diperiksa tata bahasanya semakin bertambah.

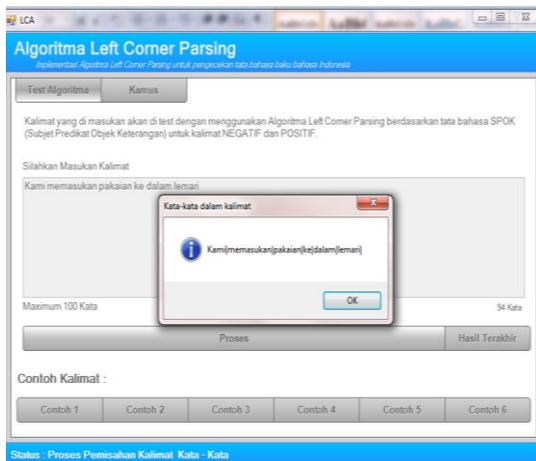
Selanjutnya hasil analisis awal ini digunakan untuk perancangan dan pembuatan aplikasi menggunakan bahasa pemrograman *Visual Basic 6.0* di mana hasil implementasinya dapat dilihat sebagai berikut:



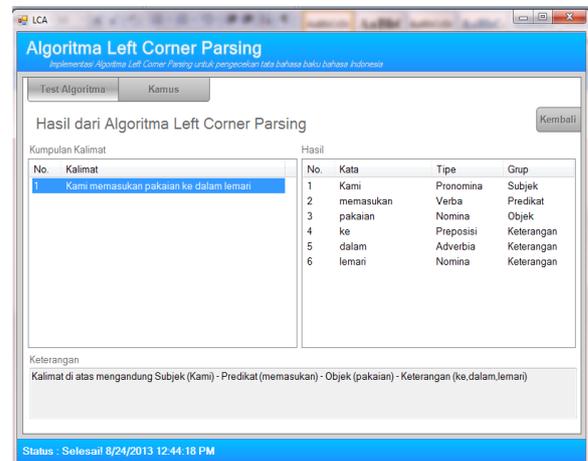
Gambar 1. Tampilan Halaman Awal



Gambar 2. Tampilan Halaman Test Algoritma



Gambar 3. Tampilan Notifikasi Proses



Gambar 4. Tampilan Halaman Hasil Test Algoritma

Setelah mengimplementasikan hasil analisis awal dan algoritma ke dalam aplikasi maka dilakukan pengujian terhadap beberapa kalimat sebagai berikut:

Tabel 1. Pengujian Kalimat

Kalimat	Kata	Jenis Kata	Hasil
Mereka berasal dari Surabaya	1.Mereka 2.Berasal 3. dari 4. Surabaya	1. Kata ganti 2. Kata keterangan 3. Kata depan 4. Kata benda	Subjek Predikat Keterangan Keterangan
Saya memasukkan pakaian ke dalam lemari	1.Saya 2.Memasukkan 3.Pakaian 4.Ke 5.Dalam 6.Lemari	1.Kata Ganti 2. Kata kerja 3. Kata benda 4. Kata depan 5. Kata keterangan 6. Kata benda	Subjek Predikat Objek Keterangan Keterangan Keterangan
Dia berasal saya	1. Dia 2. Berasal 3. Saya	1. Kata Ganti 2. Kata kerja 3. Kata Ganti	Pola kalimat tidak sesuai

Dari hasil pengujian ditemukan bahwa aplikasi dapat mengenali pola kalimat dari kalimat yang dimasukkan. Aplikasi ini memiliki ketergantungan yang tinggi terhadap pemilihan kata dan kalimat yang dimasukkan sehingga jika pemasukan kalimat tidak benar, maka akan dihasilkan pola kalimat yang tidak benar.

## V. Kesimpulan

Dari keseluruhan pembahasan yang telah dilakukan maka kesimpulan yang diperoleh adalah:

1. Algoritma *left corner parsing* dapat diterapkan pada suatu aplikasi untuk memeriksa tata bahasa dalam suatu kalimat Bahasa Indonesia.
2. Algoritma yang diterapkan pada aplikasi sangat dipengaruhi oleh pemilihan jenis kata. Jika jenis kata yang dipilih benar maka akan dapat mengenali aturan yang benar dan sebaliknya jika jenis kata yang dipilih salah maka aturan yang dikenali juga akan salah.
3. Penggunaan tanda titik dan tanda koma mempengaruhi pembagian kalimat.

## Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Fakultas Teknik Universitas Katolik De La Salle Manado yang telah memberikan sumbangan dana dalam penelitian ini.

## Daftar Pustaka

1. Abdurohman, M. Syamsul, H., Rohidin, D., 2006, "Pemeriksaan Tata Bahasa dalam Kalimat Bahasa Inggris Menggunakan Algoritma Left Corner Parsing", [http://publication.gunadarma.ac.id/bitstream/123456789/798/1/ComputationalScience\\_MamanAbdurohman\(7\)76\\_82.pdf](http://publication.gunadarma.ac.id/bitstream/123456789/798/1/ComputationalScience_MamanAbdurohman(7)76_82.pdf), diakses pada tanggal 20 Juli 2015.
2. Pandiangan, V. M., 2015. "Implementasi Left Corner Parser Pada Perancangan Aplikasi Pemeriksaan Tata Bahasa Dalam Kalimat Bahasa Indonesia", *Informasi dan Teknologi Ilmiah (INTI)*, 5, 2, 161-165.
3. Mulyono, I., 2012. *Ihwal Kalimat Bahasa Indonesia*. Bandung: CV Yrama Widya.
4. Rodriguez, C. G. 2010. *Parsing Schemata for Practical Text Analysis (Mathematics, Computing, Language, and Life:Frontiers in Mathematical)*, Volume 1, Imperial College Press, Singapore.
5. Blackburn, P., Striegnitz, K. 2002, "9.1 Left-Corner Parsing", <http://cs.union.edu/~striegnk/courses/nlp-with-prolog/html/node53.html>, diakses 28 Juli 2015.