**LAPORAN TUGAS AKHIR**

**TBO**



Disusun Oleh :

Nama : Fajri Hidayatul Ihsan

NIM : 21346007

Prodi : Informatika (NK)

Dosen Pengampu : Widya Darwin S.Pd., M.Pd.T

**PROGRAM STUDI INFORMATIKA**

**JURUSAN TEKNIK ELEKTRONIKA**

**FAKUTAS TEKNIK**

**UNIVERSITAS NEGERI PADANG**

**2023**

**Penyederhanaan aturan Produksi Context Free Grammar**

1. Tujuan Penyederhanaan

Melakukan pembatasan, sehingga tidak dihasilkan pohon penurunan yang memiliki kerumitan yang tak perlu atau aturan produksi yang tidak berarti

Misal

Diketahui suatu tata bahasa konteks:

S -> AB | a

A -> a

Kelemahan : Aturan produksi S -> AB tidak berarti karena B tidak memiliki penurunan

1. Teknik Penyederhanaan
2. Penghilangan Produksi ε

Produksi ε (Empty) adalah produksi dalam bentuk α -> ε atau bisa dianggap sebagai produksi kosong. Penghilangan produksi ε dilakukan dengan melakukan penggantian produksi yang memuat variable yang manuju produksi ε, atau biasa disebut nullable.

Contoh 1 :

Diketahui tata bahasa bebas konteks sebagai berikut :

S ->dA | Bd

A ->bc

A ->ε

B ->c

Sehingga aturan produksi setelah penyederhanaan : S ->dA | d | Bd

A ->bc

B ->c

Contoh 2 :

S -> bcAd

A ->ε

Pada kasus 1, A nullable serta A ->ε merupakan satu-satunya produksi dari A maka variable A bias ditiadakan.

Maka hasil penyederhanaan adalah : S ->bcd

Kasus 2 S ->bcAd

A ->bd | ε

Pada kasus 2, A nullable, tapi A ->ε bukan satu-satunya produksi dari A. Maka hasil penyederhanaan adalah : S ->bcAd | bcd A ->bd

1. Penghilangan Produksi Unit

Produksi unit adalah produksi yang ruas kiri dan kanan aturan produksinya hanya berupa satu simbol variable. ( α = 1 N/V dan β = 1 N/V )

Dengan adanya bentuk produksi unit ini membuat tata bahasa memiliki kerumitan yang tidak perlu atau menambah panjang penurunan.

Penyederhanaan ini dilakukan dengan melakukan penggantian aturan produksi unit.

Contoh 1 :

Diketahui tata bahasa bebas konteks sebagai berikut :

S ->Sb

S ->C

C ->D

C ->ef

D ->dd

Sehingga aturan produksi setelah penyederhanaan :

S ->Sb | dd | ef

C ->dd | ef

D ->dd

Contoh 2 :

Diketahui tata bahasa bebas konteks sebagai berikut :

S ->A

S ->Aa

A ->B

B ->C

B ->b

C ->D

C ->ab

D ->b

Sehingga aturan produksi setelah penyederhanaan :

S ->A => S ->ab | b

S ->Aa

A ->B => A ->ab | b B ->ab B ->b

C ->b

C ->ab

D ->b

1. Penghilang Produksi Useless

Produksi yang memuat simbol variable yang tidak memiliki penurunan yang akan menghasilkan terminal-terminal seluruhnya (menuju terminal), produksi ini tidak berguna karena bila diturunkan tidak akan pernah selesai (masih ada simbol variable tersisa).

Produksi yang tidak akan pernah dicapai dengan penurunan apapun dari simbol awal, sehingga produksi itu redundan (berlebih).

Contoh 1 :

Diketahui tata bahasa bebas konteks sebagai berikut

S ->aSa | Abd | Bde

A ->Ada B ->BBB | a

Maka tata bahasa hasil penyederhanaan adalah :

S ->aSa | Bde

B ->BBB | a

Contoh 2 :

Diketahui tata bahasa bebas konteks :

S ->Aa | B

A ->ab | D

B ->b | E

C ->bb

E ->aEa

Maka tata bahasa hasil penyederhana an menjadi :

S ->Aa | B

A ->ab

B ->b

1. Gabungan Useless, Unit & ε

Urutannya sebagai berikut :

1. Hilangkan produksi ε
2. Hilangkan produksi unit
3. Hilangkan produksi useless

Contoh 1 :

Hilangkan produksi useless, unit dan empty dari tata bahasa bebas konteks berikut :

S ->a | aA | B | C

A ->aB | ε

B ->Aa

C ->cCD

D ->ddd

1. Hilangkan produksi ε

S ->a | aA | B | C

A ->aB | ε

B ->Aa

C ->cCD

D ->ddd

Hasilnya

S ->a | aA | B | C

A ->aB

B ->Aa | a

C ->cCD

D ->ddd

1. Hilangkan produksi unit

S ->a | aA| B | C

A ->aB

B ->Aa | a

C ->cCD

D ->ddd

Hasilnya

S ->a | aA| Aa | cCD

A ->aB

B ->Aa | a

C ->cCD

D ->ddd

1. Hilangkan produksi useless

S ->a | aA| Aa | ~~cCD~~

A ->aB

B ->Aa | a

~~C ->cCD~~

~~D ->ddd~~

Hasilnya

S ->a | aA| Aa

A ->aB

B ->Aa | a

**Chomsky Normal Form (CNF**)

Merupakan salah satu bentuk normal yang sangat berguna untuk tata Bahasa bebas konteks (CFG)

Chomsky Normal Form (CNF) dapat dibentuk dari CFG yang telah mengalami penyederhanaan yaitu penghilangan produksi useless, unit dan kosong.

1. Bentuk Normal Chomsky

Syarat Konversi CFD ke CNF

Tidak memiliki produksi useless

Tidak memiliki Produksi unit

Tidak memiliki Produksi ε (Kosong)

Aturan Produksi CNF

Ruas kanan hanya boleh berupa sebuah simbol terminal atau dua buah simbol variable.

Jika terdapat lebih dari satu simbol terminal maka harus dilakukan penggantian dan juga jika terdapat lebih dari dua buah simbol variable maka harus dilakukan perubahan.

α ->β

α = 1 Non-Terminal

β = 1 Terminal atau 2 Non-Terminal

1. Pembentukan Bentuk Normal Chomsky

Langkah Secara Umum

1. Biarakan aturan produksi yang telah dalalm bentuk CNF
2. Lakukan penggantian aturan produksi yang ruas kanannya memuat symbol terminal dan panjang ruas kanan >1
3. Lakukan penggantian atiran produksi yang ruas kanannya memuat >2symbol variable
4. Penggantian-penggantian tersebut bisa dilakukan berkali kali sampai akhirnya semua aturan produksi dalam bentuk normal Chomsky
5. Selama dilakukan penggantian, kemungkinan kita akan memperoleh aturan-aturan produksi baru, dna juga memunculkan symbol-simbol variable baru.

Tahapan Pembentukan CNF



Contoh 1 :

Perhatikan aturan produksi CFG berikut, diasumsikan telah disederhanakan :

S → bA | aB

A → bAA | aS | a

B → aBB | bS | b

Ubahlah ke dalam bentuk normal Chomsky !!!

Jawab :

Aturan produksi yang sudah dalam bentuk normal Chomsky:

A → aB → bb.

Dilakukan penggantian aturan produksi yang belum bentuk normal Chomsky(‘=>’ bisa dibaca berubah menjadi):

S → bA => S → P1A

S → aB => S → P2B

A → bAA => A → P1AA => A → P1P3

A → aS => A → P2S

B → aBB => B → P2BB => B → P2P4

B → bS => B → P1Sc.

Terbentuk aturan produksi dan simbol variabel baru:

P1 → b

P2 → a

P3 → AA

P4 → BBd.

Hasil akhir aturan produksi dalam bentuk normal Chomsky :

A → a

B → b

S → P1A

S → P2B

A → P1P3

A → P2S

B → P2P4

B → P1S

P1 → b

P2 → a

P3 → AA

P4 → BB

Contoh 2 :

Berikut adalah tata bahasa bebas konteks yang sudah disederhanakan.

S -> aB | CA

A -> a | bc B -> BC | Ab

C -> aB | b Dari aturan produksi diatas, aturan produksi yang sudah dalam bentuk normal Chomsky (CNF) adalah: S -> CA

A -> a

B -> BC

C -> b

Aturan produksi yang belum dalam CNF adalah:

S -> aB => S -> P1B

A -> bc =>A -> P2P3

B -> Ab => B -> AP2

C -> aB =>C -> P1B

Terbentu aturan produksi baru:

P1 -> a

P2 -> b

P3 -> c

Hasil ahir aturan produksi dalam CNF adalah:

S -> CA

A -> a

B -> BC

C -> b

S -> P1B

A -> P2P3

B -> AP2

C -> P1B

P1 -> a

P2 -> b

P3 -> c

1. Algoritma CYK untuk CFG

Algoritma the Cocke-Younger-Kasami algorithm (CYK) merupakan algoritma parsing dan keanggotaan (Membership) untuk tata Bahasa Bebas Konteks. Algoritma ini diciptakan oleh J.Cocke.DH.Younger. dan T.Kasami. Syarat untuk menggunakan algoritma ini adalah tata bahasa harus sudah dalam bentuk Normal Chomsky

Simbol Algoritma CYK

n = panjang string yang akan diperiksa, misalnya 'ada', n = ladal =3

I menyatakan kolom ke-

J menyatakan baris ke-

Tahapan no (2) dan (3) untuk mengisi tabel baris pertama kolom 1-n

No (4), iterasi dari baris ke-2 sampai no (4)

No (5), iterasi untuk mengisi kolom 1 sampai (n-baris+1) pada suatu baris

No(7), inisialisasi Vij dengan Ø

No (8) dan (9), iterasi untuk memeriksa mana saja yang menjadi anggota Vij

Contoh 1 :

Diketahui tata bahasa bebas konteks:

S -> AB | BC

A ->BA | a

B -> CC | b

C -> AB | a

Periksa, apakah untai ‘baaba’ termasuk ke dalam bahasa tersebut.

Penyelesaian:

Syarat agar sebuah untai termasuk ke dalam tata bahasa tertentu adalah V1n

harus memuat simbol awal

i -> ( 1 – 5)

j -> 1 – 5)

Hasil pada tabel

