

LAPORAN TUGAS PEMROGRAMAN
IF2124 TEORI BAHASA FORMAL DAN OTOMATA
HTML CHECKER DENGAN PUSHDOWN AUTOMATA (PDA)

Kelompok 22 “Judi Slot Gacor”

Ahmad Mudabbir Arif 13522072

Muhammad Neo Cicero Koda 13522108

William Glory Henderson 13522113



SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

2023

DAFTAR ISI

LAPORAN TUGAS PEMROGRAMAN.....	1
DAFTAR ISI.....	2
BAB I.....	3
BAB II.....	4
2.1. Teori dasar PDA.....	4
2.1.1. Transisi PDA.....	4
2.1.2. Operasi Stack.....	5
2.1.3. Bahasa Diterima oleh PDA.....	5
2.1.4. Deterministik dan Non Deterministik.....	5
2.1.5. Hubungan dengan Bahasa Context-Free.....	5
2.2. Pengaplikasian PDA pada Sintaksis HTML.....	5
2.2.1. HTML Sebagai Bahasa Context-Free.....	5
2.2.2. Input dan Stack Symbols.....	6
2.2.3. Transisi State.....	6
2.2.4. Hierarki Tumpukan.....	6
2.2.5. Parsing HTML.....	6
BAB III.....	7
3.1. Hasil PDA yang dibuat.....	7
3.2. Implementasi dan Pengujian.....	20
3.2.1. Deskripsi Program Utama.....	20
3.2.2. Spesifikasi teknis program.....	21
BAB IV.....	24
4.1. Contoh 1 (test.html).....	24
4.2. Contoh 2 (test2.html).....	24
4.3. Contoh 3 (test3.html).....	25
4.4. Contoh 4 (test4.html).....	25
BAB V.....	27
BAB VI.....	28

BAB I

Latar Belakang

HTML (Hypertext Markup Language) adalah bahasa markup yang digunakan untuk membuat struktur dan tampilan konten web. HTML adalah salah satu bahasa utama yang digunakan dalam pengembangan web dan digunakan untuk menggambarkan bagaimana elemen-elemen konten, seperti teks, gambar, tautan, dan media, akan ditampilkan di browser web. Setiap dokumen HTML dimulai dengan elemen `<html>`, lalu diikuti dengan `<head>` (untuk metadata dan tautan ke file eksternal) dan `<body>` (untuk konten yang akan ditampilkan)

HTML menggunakan elemen-elemen (*tags*) untuk mengelompokkan dan mengatur konten. Contohnya, `<p>` digunakan untuk paragraf teks, `<h1>` hingga `<h6>` digunakan untuk judul, `<a>` untuk tautan, `` untuk gambar, dan sebagainya. Elemen HTML sering memiliki atribut yang memberikan informasi tambahan tentang elemen tersebut. Contohnya adalah atribut `src` untuk gambar, `href` untuk tautan, dan `class` untuk memberikan elemen kelas CSS.

Sama seperti bahasa pada umumnya, HTML juga memiliki sintaks tersendiri dalam penulisannya yang dapat menimbulkan error jika tidak dipenuhi. Meskipun web browser modern seperti Chrome dan Firefox cenderung tidak menghiraukan error pada HTML memastikan bahwa HTML benar dan terbentuk dengan baik masih penting untuk beberapa alasan seperti *Search Engine Optimization (SEO)*, aksesibilitas, *maintenance* yang lebih baik, kecepatan render, dan profesionalisme.

Dibutuhkan sebuah program pendeteksi *error* untuk HTML. Oleh sebab itu, diimplementasikanlah sebuah program yang dapat memeriksa kebenaran HTML dari segi nama *tag* yang digunakan serta *attribute* yang dimilikinya. Digunakan konsep Pushdown Automata (PDA) dalam mencapai hal tersebut yang diimplementasikan dalam bahasa Python.

BAB II

Teori Dasar

2.1. Teori dasar PDA

PDA (Pushdown Automaton) adalah salah satu model mesin abstrak dalam teori bahasa formal dan otomata. PDA merupakan otomaton yang lebih kuat daripada DFA (Deterministic Finite Automaton) dan NFA (Nondeterministic Finite Automaton) karena memiliki kemampuan untuk menggunakan *stack* (tumpukan) sebagai memori tambahan. PDA merupakan *7-tuple* yang memiliki komponen-komponen sebagai berikut:

$$P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$$

1. *State* (Q): Sekumpulan *state* sehingga yang menggambarkan keadaan PDA pada saat itu.
2. *Input Alphabet* (Σ): Sekumpulan simbol yang dapat diterima sebagai masukan.
3. *Stack Alphabet* (Γ): Sekumpulan simbol yang dapat ditempatkan (*push*) atau dikeluarkan (*pop*) dari tumpukan.
4. *Transition Function* (δ): Aturan yang menentukan bagaimana PDA bertransisi dari sebuah *state* ke *state* lain berdasarkan simbol input dan simbol pada bagian paling atas tumpukan (*top of stack*).
5. *Start State* (q_0): State awal ketika PDA memproses masukan.
6. *Final States* (F): Kumpulan *state* yang menandakan bahwa masukan diterima jika masukan sudah habis dan PDA dapat mencapai *state* tersebut.
7. *Stack* (Z_0 , *Start symbol*): Tumpukan yang digunakan untuk menyimpan simbol. Sebuah *stack* hanya dapat diakses dengan memproses simbol teratasnya (*top of stack*).

PDA mengakses simbol-simbol dari input dan tumpukan, dan berpindah antar state sesuai dengan aturan yang ditentukan oleh fungsi transisi. PDA dapat berpindah ke beberapa state pada saat yang sama dan menggunakan tumpukan untuk menyimpan informasi sementara.

2.1.1. Transisi PDA

Transisi PDA terjadi berdasarkan *state* yang dia berada pada saat ini, simbol *input*, dan simbol tumpukan saat ini. Fungsi transisi δ dapat bersifat non deterministik, yang berarti dari satu *state* dan konfigurasi tertentu, PDA dapat berpindah ke beberapa *state* dan konfigurasi.

2.1.2. Operasi *Stack*

PDA menggunakan *stack* (tumpukan) untuk menyimpan dan mengakses informasi. Operasi utama pada tumpukan adalah *push* (menambahkan simbol ke tumpukan) dan *pop* (menghapus simbol dari tumpukan).

2.1.3. Bahasa Diterima oleh PDA

Bahasa diterima oleh PDA adalah himpunan *string* yang dapat diterima oleh PDA sesuai dengan aturan transisi yang didefinisikan. Penerimaan sebuah *string* oleh PDA dapat terjadi melalui tumpukan kosong atau melalui *final state*. PDA dapat menerima bahasa konteks-bebas (*Context-Free Languages*) yang lebih kuat daripada bahasa yang dapat diterima oleh DFA.

2.1.4. Deterministik dan Non Deterministik

PDA dapat bersifat deterministik atau non deterministik tergantung pada apakah fungsi transisinya memungkinkan beberapa transisi dari satu keadaan dan konfigurasi tertentu.

2.1.5. Hubungan dengan Bahasa *Context-Free*

PDA dapat digunakan untuk mengenali dan memproses bahasa konteks-bebas. Terdapat teorema yang menunjukkan bahwa setiap bahasa konteks-bebas dapat diterima oleh PDA.

Pemahaman teori dasar tersebut penting untuk memahami konsep dasar dalam teori bahasa formal dan otomata serta untuk merancang PDA yang dapat mengenali bahasa tertentu. PDA memiliki peran penting dalam analisis sintaksis dalam kompilasi dan pemrosesan bahasa alami.

2.2. Pengaplikasian PDA pada Sintaksis HTML

2.2.1. HTML Sebagai Bahasa *Context-Free*

HTML dapat dianggap sebagai bahasa *Context-Free* karena struktur dan sintaksis yang dimilikinya dapat didefinisikan oleh *Context-Free Grammar* (CFG). Komponen-komponen pembangun utama dalam HTML merupakan *tag-tag* bersarang yang bersifat hierarkis. Struktur tersebut memungkinkan HTML untuk dapat diproses dengan CFG. Karena PDA bersifat ekuivalen dengan CFG, HTML dapat diproses dengan menggunakan PDA.

2.2.2. Input dan Stack Symbols

Sebelum diproses oleh PDA, sebuah *file* HTML dapat melewati proses tokenisasi terlebih dahulu agar PDA tidak memproses *file* secara karakter-per-karakter. Simbol input dapat mewakili token HTML seperti tag pembuka, tag penutup, atribut, dan teks. Simbol tumpukan mewakili struktur hierarki HTML, seperti tag pembuka, tag penutup, atau atribut.

2.2.3. Transisi *State*

Transisi antar *state* pada PDA dapat mewakili perubahan dalam parsing HTML. Misalnya, transisi ke *state* baru setelah membaca tag pembuka atau tag penutup. Contoh lainnya adalah transisi dari *state* pembangunan *tag* head ke *state* pemrosesan isi head. Transisi tersebut menyebabkan perubahan dalam masukan yang dapat diterima oleh PDA.

2.2.4. Hierarki Tumpukan

PDA menggunakan tumpukan untuk memahami dan memelihara hierarki elemen HTML. Tag pembuka dimasukkan ke dalam tumpukan, dan tag penutup dikeluarkan saat tag yang sesuai ditemukan. Sebagai contoh, jika pada atas tumpukan terdapat token “div”, hal tersebut berarti bahwa pada saat itu, PDA sedang memproses bagian yang berada di dalam tag div. Transisi selanjutnya yang dapat dilakukan oleh PDA adalah transisi dengan *top of stack* berupa token “head”.

2.2.5. Parsing HTML

PDA dapat digunakan untuk merinci proses parsing HTML dan memastikan bahwa struktur dokumen HTML mematuhi aturan sintaksis yang benar. Hal tersebut membantu dalam menghindari kesalahan sintaksis dalam pengembangan *web*. Memastikan bahwa tidak ada kesalahan sintaksis yang muncul dapat berdampak baik kepada beberapa aspek, seperti *Search Engine Optimization* (SEO) dan *maintenance* yang lebih baik.

BAB III

Hasil dan Implementasi

3.1. Hasil PDA yang dibuat

Hasil PDA yang dibuat terdapat pada file pda.txt dengan hasil sebagai berikut.

[current state] [input word] [top of stack] [next state] [element to push]

```
S0 S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 S11 S12 S13 S14 S15 S16 S17 S18
S19 S20 S21 S22 S23 S24 S25 S26 S27 S28 S29 S30 S31 S32 S33 S34 S35
S36 S37 S38 S39 S40 S41 S42 S43 S44 S45 S46 S47 S48 S49 S50 S51 S52
S53 S54 S55 S56 S57 S58 S59 S60 S61 S62 S63 S64 S65 S66 S67 S68 S69
S70 S71 S72 S73 S74 S75 S76 S77 S78 S79
< > html head body </ title any anyattr link href= rel= alt= " '
script src= h1 h2 h3 h4 h5 h6 table tr th td br div em b abbr
strong small a hr img form method= action= GET POST button type=
submit reset input password text email number checkbox id= class=
style= <!-- -->
Z < > html head body </ title any anyattr link href= rel= alt= " '
script src= h1 h2 h3 h4 h5 h6 table tr th td br div em b abbr
strong small a hr img form method= action= GET POST button type=
submit reset input password text email number checkbox id= class=
style=
<!-- -->
S0
Z
S4
F
S0 < Z S0 <Z
S0 html < S0 html<
S0 > html S1 html
S1 < html S1 <html
S1 head < S1 head<
S1 > head S2 head
S2 </ head S3 head
S2 e head S14 head
S3 head head S3 e
S3 > < S3 e
S3 e html S5 html
S4 html html S4 e
S4 > < S4 e
S5 < html S5 <html
S5 body < S5 body<
S5 > body S6 body
S6 any div S6 div
S6 any form S6 form
S6 </ body S7 body
S6 e body S8 body
```

```
S6 e div S8 div
S6 < body S35 body
S6 < div S35 div
S6 </ div S37 div
S6 </ form S37 form
S6 e body S33 body
S6 e div S33 div
S6 < body S41 body
S6 < div S41 div
S6 < form S41 form
S6 e body S14 body
S6 e div S14 div
S6 e form S8 form
S6 e form S14 form
S6 e form S33 form
S6 < form S35 form
S7 body body S7 e
S7 > < S7 e
S7 </ html S4 html
S8 < body S8 <body
S8 p < S8 p<
S8 h1 < S8 h1<
S8 h2 < S8 h2<
S8 h3 < S8 h3<
S8 h4 < S8 h4<
S8 h5 < S8 h5<
S8 h6 < S8 h6<
S8 table < S8 table<
S8 > p S9 p
S8 > h1 S9 h1
S8 > h2 S9 h2
S8 > h3 S9 h3
S8 > h4 S9 h4
S8 > h5 S9 h5
S8 > h6 S9 h6
S8 > table S21 table
S8 < div S8 <div
S8 < form S8 <form
S9 any p S9 p
S9 any h1 S9 h1
S9 any h2 S9 h2
S9 any h3 S9 h3
S9 any h4 S9 h4
S9 any h5 S9 h5
S9 any h6 S9 h6
S9 </ p S10 p
S9 </ h1 S10 h1
S9 </ h2 S10 h2
S9 </ h3 S10 h3
S9 </ h4 S10 h4
S9 </ h5 S10 h5
```



```
S9 </ h6 S10 h6
S9 < p S32 p
S9 < h1 S32 h1
S9 < h2 S32 h2
S9 < h3 S32 h3
S9 < h4 S32 h4
S9 < h5 S32 h5
S9 < h6 S32 h6
S9 e p S33 p
S9 e h1 S33 h1
S9 e h2 S33 h2
S9 e h3 S33 h3
S9 e h4 S33 h4
S9 e h5 S33 h5
S9 e h6 S33 h6
S10 p p S10 e
S10 h1 h1 S10 e
S10 h2 h2 S10 e
S10 h3 h3 S10 e
S10 h4 h4 S10 e
S10 h5 h5 S10 e
S10 h6 h6 S10 e
S10 > < S10 e
S10 e body S6 body
S10 e div S6 div
S10 e form S6 form
S11 src= script S11 src=script
S11 " src= S47 "src=
S11 e script S19 script
S12 any title S12 title
S12 </ title S13 title
S13 title title S13 e
S13 > < S13 e
S13 e head S2 head
S14 < head S14 <head
S14 link < S14 link<
S14 title < S14 title<
S14 script < S14 script<
S14 e link S15 link
S14 e script S11 script
S14 > title S12 title
S14 < body S14 <body
S14 < div S14 <div
S15 href= link S15 href=link
S15 rel= link S15 rel=link
S15 e href= S48 href=
S15 e rel= S48 rel=
S16 e link S15 link
S16 e href= S16 e
S16 e rel= S17 e
S17 e link S18 e
```

```
S18 e head S2 head
S18 > < S18 e
S18 e body S6 body
S18 e div S6 div
S19 > script S49 script
S19 e script S61 script
S20 script script S20 e
S20 > < S20 e
S20 e head S2 head
S20 e body S6 body
S20 e div S6 div
S21 < table S21 <table
S21 tr < S21 tr<
S21 > tr S22 tr
S21 e table S30 table
S22 < tr S22 <tr
S22 th < S22 th<
S22 > th S23 th
S22 e tr S29 tr
S23 any th S23 th
S23 </ th S24 th
S23 e th S44 th
S33 br th S33 brth
S33 > br S23 e
S24 e tr S22 tr
S24 th th S24 e
S24 > < S24 e
S24 </ tr S25 tr
S25 tr tr S25 e
S25 > < S25 e
S25 </ table S31 table
S25 e table S26 table
S26 < table S26 <table
S26 tr < S26 tr<
S26 > tr S27 tr
S27 < tr S27 <tr
S27 td < S27 td<
S27 > td S28 td
S27 e tr S29 tr
S28 any td S28 td
S28 </ td S29 td
S28 e td S44 td
S29 td td S29 e
S29 > < S29 e
S29 > < S27 e
S29 </ tr S30 tr
S30 tr tr S30 e
S30 > < S30 e
S30 </ table S31 table
S30 e table S26 table
S31 table table S31 e
```

```
S31 > < S31 e
S31 e body S6 body
S31 e div S6 div
S31 e form S6 form
S32 br p S32 brp
S32 br h1 S32 brh1
S32 br h2 S32 brh2
S32 br h3 S32 brh3
S32 br h4 S32 brh4
S32 br h5 S32 brh5
S32 br h6 S32 brh6
S32 > br S9 e
S33 < body S33 <body
S33 < div S33 <div
S33 < form S33 <form
S33 < p S33 <p
S33 < h1 S33 <h1
S33 < h2 S33 <h2
S33 < h3 S33 <h3
S33 < h4 S33 <h4
S33 < h5 S33 <h5
S33 < h6 S33 <h6
S33 em < S33 em<
S33 b < S33 b<
S33 abbr < S33 abbr<
S33 strong < S33 strong<
S33 small < S33 small<
S33 > em S34 em
S33 > b S34 b
S33 > abbr S34 abbr
S33 > strong S34 strong
S33 > small S34 small
S33 a < S33 a<
S33 > a S34 a
S33 href= a S42 href=a
S34 any em S34 em
S34 any b S34 b
S34 any abbr S34 abbr
S34 any strong S34 strong
S34 any small S34 small
S34 < em S40 em
S34 < b S40 b
S34 < abbr S40 abbr
S34 < strong S40 strong
S34 < small S40 small
S34 </ em S39 em
S34 </ b S39 b
S34 </ abbr S39 abbr
S34 </ strong S39 strong
S34 </ small S39 small
S34 any a S34 a
```

```
S34 </ a S39 a
S35 div body S51 divbody
S35 div div S51 divdiv
S35 div form S51 divform
S35 form body S52 formbody
S35 form div S52 formdiv
S37 div div S38 e
S37 form form S38 e
S38 > body S6 body
S38 > div S6 div
S38 > form S6 form
S39 e p S9 p
S39 e h1 S9 h1
S39 e h2 S9 h2
S39 e h3 S9 h3
S39 e h4 S9 h4
S39 e h5 S9 h5
S39 e h6 S9 h6
S39 em em S39 e
S39 b b S39 e
S39 abbr abbr S39 e
S39 strong strong S39 e
S39 small small S39 e
S39 a a S39 e
S39 > < S39 e
S39 e body S6 body
S39 e div S6 div
S39 e th S23 th
S39 e td S28 td
S39 e form S6 form
S40 br em S40 brem
S40 br b S40 brb
S40 br abbr S40 brabbr
S40 br strong S40 brstrong
S40 br small S40 brsmall
S40 > br S34 e
S41 hr body S41 hrbody
S41 hr div S41 hrdiv
S41 hr form S41 hrform
S41 br body S41 brbody
S41 br div S41 brdiv
S41 br form S41 brform
S41 > hr S6 e
S41 > br S6 e
S41 img body S41 imgbody
S41 img div S41 imgdiv
S41 img form S41 imgform
S41 e img S45 img
S42 " href= S42 "href=
S42 anyattr " S42 "
S42 " " S43 e
```

```
S43 e href= S33 e
S44 < th S44 <th
S44 < td S44 <td
S44 em < S44 em<
S44 b < S44 b<
S44 abbr < S44 abbr<
S44 strong < S44 strong<
S44 small < S44 small<
S44 > em S34 em
S44 > b S34 b
S44 > abbr S34 abbr
S44 > strong S34 strong
S44 > small S34 small
S45 alt= img S45 alt=img
S45 src= img S45 src=img
S45 " alt= S45 "alt=
S45 " src= S45 "src=
S45 anyattr " S45 "
S45 " " S46 e
S46 e alt= S46 e
S46 e img S45 img
S46 e src= S50 e
S47 anyattr " S47 "
S47 " " S47 e
S47 e src= S19 e
S48 " href= S48 "href=
S48 " rel= S48 "rel=
S48 anyattr " S48 "
S48 " " S16 e
S49 any script S49 script
S49 </ script S20 script
S50 > img S6 e
S51 > div S6 div
S52 action= form S52 form
S52 method= form S52 method=form
S52 " form S53 "form
S52 " method= S54 method=
S52 > form S6 form
S53 anyattr " S53 "
S53 anyattr ' S53 '
S53 ' ' S55 e
S53 " " S55 e
S54 POST method= S56 method=
S54 GET method= S56 method=
S55 e form S52 form
S55 > form S6 form
S56 " method= S55 e
S41 input body S41 inputbody
S41 input div S41 inputdiv
S41 input form S41 inputform
S41 type= input S57 input
```

```
S41 e input S58 input
S57 " input S57 "input
S57 text " S58 "
S57 password " S58 "
S57 number " S58 "
S57 checkbox " S58 "
S57 email " S58 "
S58 " " S58 e
S58 > input S6 e
S8 button < S8 button<
S8 > button S9 button
S9 any button S9 button
S9 </ button S10 button
S10 button button S10 e
S8 type= button S59 button
S9 e button S33 button
S33 < button S33 <button
S39 e button S9 button
S59 " button S59 "button
S59 submit " S60 "
S59 reset " S60 "
S59 button " S60 "
S60 " " S60 e
S60 > button S9 button
S60 e button S8 button
S0 id= html S61 html
S0 class= html S61 html
S0 style= html S61 html
S1 id= head S61 head
S1 class= head S61 head
S1 style= head S61 head
S5 id= body S61 body
S5 class= body S61 body
S5 style= body S61 body
S14 id= script S61 script
S14 class= script S61 script
S14 style= script S61 script
S14 id= title S61 title
S14 class= title S61 title
S14 style= title S61 title
S14 id= link S61 link
S14 class= link S61 link
S14 style= link S61 link
S15 id= link S61 link
S15 class= link S61 link
S15 style= link S61 link
S19 id= script S61 script
S19 class= script S61 script
S19 style= script S61 script
S61 " script S61 "script
S61 " link S61 "link
```

```
S61 " title S61 "title
S61 " head S61 "head
S61 " body S61 "body
S61 " html S61 "html
S61 " hr S61 "hr
S61 " img S61 "img
S61 " input S61 "input
S61 anyattr " S61 "
S61 " " S62 e
S62 e html S0 html
S62 e head S1 head
S62 e body S5 body
S62 e script S14 script
S62 e link S14 link
S62 e title S14 title
S62 e input S41 input
S62 e img S41 img
S62 e hr S41 hr
S62 e img S45 img
S41 id= hr S61 hr
S41 class= hr S61 hr
S41 style= hr S61 hr
S41 id= img S61 img
S41 class= img S61 img
S41 style= img S61 img
S41 id= input S61 input
S41 class= input S61 input
S41 style= input S61 input
S58 id= input S61 input
S58 class= input S61 input
S58 style= input S61 input
S45 id= input S61 input
S45 class= input S61 input
S45 style= input S61 input
S51 id= div S63 div
S51 class= div S63 div
S51 style= div S63 div
S52 id= form S63 form
S52 class= form S63 form
S52 style= form S63 form
S63 " form S63 "form
S63 " div S63 "div
S63 anyattr " S63 "
S63 " " S64 e
S64 e form S52 form
S64 e div S51 div
S8 id= h1 S65 h1
S8 id= h2 S65 h2
S8 id= h3 S65 h3
S8 id= h4 S65 h4
S8 id= h5 S65 h5
```

```
S8 id= h6 S65 h6
S8 id= p S65 p
S8 id= button S65 button
S8 id= table S65 table
S8 class= h1 S65 h1
S8 class= h2 S65 h2
S8 class= h3 S65 h3
S8 class= h4 S65 h4
S8 class= h5 S65 h5
S8 class= h6 S65 h6
S8 class= p S65 p
S8 class= button S65 button
S8 class= table S65 table
S8 style= h1 S65 h1
S8 style= h2 S65 h2
S8 style= h3 S65 h3
S8 style= h4 S65 h4
S8 style= h5 S65 h5
S8 style= h6 S65 h6
S8 style= p S65 p
S8 style= button S65 button
S8 style= table S65 table
S65 " h1 S65 "h1
S65 " h1 S65 "h1
S65 " h2 S65 "h2
S65 " h3 S65 "h3
S65 " h4 S65 "h4
S65 " h5 S65 "h5
S65 " h6 S65 "h6
S65 " p S65 "p
S65 " button S65 "button
S65 " a S65 "a
S65 " strong S65 "strong
S65 " small S65 "small
S65 " em S65 "em
S65 " br S65 "br
S65 " abbr S65 "abbr
S65 " table S65 "table
S65 anyattr " S65 "
S65 " " S66 e
S66 e h1 S8 h1
S66 e h2 S8 h2
S66 e h3 S8 h3
S66 e h4 S8 h4
S66 e h5 S8 h5
S66 e h6 S8 h6
S66 e p S8 p
S66 e button S8 button
S66 e table S8 table
S66 e b S33 b
S33 id= a S65 a
```



```
S33 id= strong S65 strong
S33 id= br S65 br
S33 id= abbr S65 abbr
S33 id= small S65 small
S33 id= em S65 em
S33 class= a S65 a
S33 class= strong S65 strong
S33 class= br S65 br
S33 class= abbr S65 abbr
S33 class= small S65 small
S33 class= em S65 em
S33 style= a S65 a
S33 style= strong S65 strong
S33 style= br S65 br
S33 style= abbr S65 abbr
S33 style= small S65 small
S33 style= em S65 em
S66 e abbr S33 abbr
S66 e em S33 em
S66 e strong S33 strong
S66 e small S33 small
S66 e a S33 a
S21 id= tr S67 tr
S21 class= tr S67 tr
S21 style= tr S67 tr
S22 id= th S67 th
S22 class= th S67 th
S22 style= th S67 th
S67 " tr S67 "tr
S67 " th S67 "th
S67 anyattr " S67 "
S67 " " S68 e
S68 e tr S21 tr
S68 e th S22 th
S26 id= tr S69 tr
S26 class= tr S69 tr
S26 style= tr S69 tr
S27 id= td S69 td
S27 class= td S69 td
S27 style= td S69 td
S69 " tr S69 "tr
S69 " td S69 "td
S69 anyattr " S69 "
S69 " " S70 e
S70 e tr S26 tr
S70 e td S27 td
S0 <!-- Z S71 Z
S1 <!-- html S71 html
S2 <!-- head S71 head
S6 <!-- body S71 body
S6 <!-- div S71 div
```

```
S71 any Z S71 Z
S71 any head S71 head
S71 any body S71 body
S71 any html S71 html
S71 any div S71 div
S71 --> Z S0 Z
S71 --> html S1 html
S71 --> head S2 head
S71 --> body S6 body
S71 --> div S6 div
S3 <!-- html S72 html
S72 any html S72 html
S72 --> html S3 html
S4 <!-- Z S73 Z
S7 <!-- html S73 html
S73 any Z S73 Z
S73 any html S73 html
S73 --> Z S4 Z
S73 --> html S7 html
S17 href= link S74 link
S17 rel= link S74 link
S74 " link S74 "link
S74 anyattr " S74 "
S74 " " S17 e
S17 id= link S76 link
S17 class= link S76 link
S17 style= link S76 link
S76 " link S76 "link
S76 anyattr " S76 "
S76 " " S77 e
S77 e link S17 link
S50 src= img S75 img
S50 alt= img S75 img
S75 " img S75 "img
S75 anyattr " S75 "
S75 " " S50 e
S50 id= img S78 img
S50 class= img S78 img
S50 style= img S78 img
S78 " img S78 "img
S78 anyattr " S78 "
S78 " " S79 e
S79 e img S50 img
S33 class= a S42 class=a
S33 id= a S42 id=a
S33 style= a S42 style=a
S42 " id= S42 "id=
S42 " class= S42 "class=
S42 " style= S42 "style=
S11 ' src= S47 'src=
S42 ' href= S42 'href=
```

```
S42 anyattr ' S42 '  
S42 ' ' S43 e  
S45 ' alt= S45 'alt=  
S45 ' src= S45 'src=  
S45 anyattr ' S45 '  
S45 ' ' S46 e  
S47 anyattr ' S47 '  
S47 ' ' S47 e  
S48 ' href= S48 'href=  
S48 ' rel= S48 'rel=  
S48 anyattr ' S48 '  
S48 ' ' S16 e  
S52 ' action= S53 action=  
S52 ' method= S54 method=  
S56 ' method= S55 e  
S57 ' input S57 'input  
S57 text ' S58 '  
S57 password ' S58 '  
S57 number ' S58 '  
S57 checkbox ' S58 '  
S57 email ' S58 '  
S58 ' ' S58 e  
S59 ' button S59 'button  
S59 submit ' S60 '  
S59 reset ' S60 '  
S59 button ' S60 '  
S60 ' ' S60 e  
S61 ' script S61 'script  
S61 ' link S61 'link  
S61 ' title S61 'title  
S61 ' head S61 'head  
S61 ' body S61 'body  
S61 ' html S61 'html  
S61 ' hr S61 'hr  
S61 ' img S61 'img  
S61 ' input S61 'input  
S61 anyattr ' S61 '  
S61 ' ' S62 e  
S63 ' form S63 'form  
S63 ' div S63 'div  
S63 anyattr ' S63 '  
S63 ' ' S64 e  
S65 ' h1 S65 'h1  
S65 ' h1 S65 'h1  
S65 ' h2 S65 'h2  
S65 ' h3 S65 'h3  
S65 ' h4 S65 'h4  
S65 ' h5 S65 'h5  
S65 ' h6 S65 'h6  
S65 ' p S65 'p  
S65 ' button S65 'button
```

```

S65 ' a S65 'a
S65 ' strong S65 'strong
S65 ' small S65 'small
S65 ' em S65 'em
S65 ' br S65 'br
S65 ' abbr S65 'abbr
S65 ' table S65 'table
S65 anyattr ' S65 '
S65 ' ' S66 e
S67 ' tr S67 'tr
S67 ' th S67 'th
S67 anyattr ' S67 '
S67 ' ' S68 e
S69 ' tr S69 'tr
S69 ' td S69 'td
S69 anyattr ' S69 '
S69 ' ' S70 e
S74 ' link S74 'link
S74 anyattr ' S74 '
S74 ' ' S17 e
S76 ' link S76 'link
S76 anyattr ' S76 '
S76 ' ' S77 e
S75 ' img S75 'img
S75 anyattr ' S75 '
S75 ' ' S50 e
S78 ' img S78 'img
S78 anyattr ' S78 '
S78 ' ' S79 e
S42 ' id= S42 'id=
S42 ' class= S42 'class=
S42 ' style= S42 'style=

```

3.2. Implementasi dan Pengujian

3.2.1. Deskripsi Program Utama

Program utama berada pada file checker.py, program ini menggunakan bahasa pemrograman Python yang digunakan untuk mengenali dan memproses bahasa HTML dengan menggunakan Pushdown Automaton (PDA). PDA adalah model otomata yang memiliki tumpukan sebagai elemen penyimpanan tambahan untuk memproses bahasa yang memiliki struktur hierarkis, seperti HTML.

Program ini menerima dua argumen dari baris perintah: nama file PDA dan nama file HTML yang akan dianalisis. Pada awal eksekusi, program membaca

aturan produksi PDA dari file PDA (pda.txt) dan menginisialisasi variabel-variabel global yang diperlukan.

Fungsi tokenize digunakan untuk memecah input HTML menjadi token-token berdasarkan aturan token yang telah ditentukan. Token-token ini kemudian digunakan dalam fungsi generate untuk membentuk konfigurasi PDA. Fungsi ini melakukan pencarian melalui state-state PDA dan membangun konfigurasi dengan mengikuti aturan produksi yang sesuai.

Fungsi parse_file bertanggung jawab membaca file PDA, menginisialisasi variabel-variabel PDA, dan membangun aturan produksi berdasarkan isi file tersebut.

Program juga mencakup fungsi done yang mencetak hasil akhir apakah PDA menerima input HTML atau tidak. Hasil akhir ini bergantung pada apakah konfigurasi akhir yang dihasilkan oleh PDA diterima atau tidak.

Selanjutnya, program memproses file HTML yang diberikan sebagai input, melakukan tokenisasi, dan memulai proses generasi konfigurasi PDA dengan memanggil fungsi generate. Output dari program akan mencetak "Accepted" jika PDA menerima input HTML, dan "Syntax Error" jika tidak serta letak kesalahan sintaksis pada *file* HTML.

3.2.2. Spesifikasi teknis program

a. Variabel Global

No.	Variabel	Deskripsi
1.	start_input	Variabel string untuk menyimpan input HTML.
2.	input_array	Array yang berisi token-token hasil tokenisasi dari input HTML.
3.	found	Variabel boolean untuk menandakan apakah PDA telah menemukan konfigurasi yang diterima.
4.	accepted_config	Array untuk menyimpan konfigurasi diterima oleh PDA
5.	productions	Dictionary untuk menyimpan aturan produksi PDA
6.	states	Array untuk menyimpan state-state PDA

7.	symbols	Array untuk menyimpan simbol-simbol input PDA
8.	stack_symbols	Array untuk menyimpan simbol-simbol tumpukan PDA
9.	start_symbol	Variabel string untuk menyimpan simbol awal PDA
10.	stack_start	Variabel string untuk menyimpan simbol awal tumpukan PDA
11.	acceptable_states	Array untuk menyimpan state-state akhir PDA.
12.	accept_with	String untuk menentukan apakah PDA menerima dengan atau tanpa tumpukan.

b. List of Tokens, daftar token yang digunakan untuk tokenisasi HTML

- '</'
- '-->'
- '<!--'
- '<'
- '>'
- 'html'
- 'head'
- 'body'
- 'alt='
- 'any'
- 'title'
- 'link'
- 'href='
- 'rel='
- ''''
- '''''
- 'script'
- 'src='
- 'h1'
- 'h2'
- 'h3'
- 'h4'
- 'h5'
- 'h6'
- 'table'
- 'tr'
- 'th'
- 'td'
- 'br'
- 'div'
- 'abbr'
- 'strong'
- 'small'
- 'hr'
- 'img'
- 'form'
- 'method='
- 'action='
- 'POST'
- 'GET'
- 'button'
- 'type='
- 'submit'
- 'reset'
- 'input'
- 'text'
- 'password'
- 'email'
- 'number'
- 'checkbox'
- 'p'
- 'b'
- 'a'
- 'em'
- 'id='
- 'class='
- 'style='

c. Fungsi dan Prosedur

No.	Fungsi/Prosedur	Deskripsi
1.	<code>tokenize(input)</code>	Fungsi untuk melakukan tokenisasi pada input HTML. Input: String input HTML. Output: Array token hasil tokenisasi.
2.	<code>tokenize_pda(input)</code>	Fungsi untuk melakukan tokenisasi pada PDA. Input: String input PDA. Output: Array token hasil tokenisasi.
3.	<code>generate(state, input, stack, config)</code>	Fungsi rekursif untuk menghasilkan konfigurasi PDA. Input: State saat ini, input saat ini, tumpukan saat ini, konfigurasi saat ini. Output: Total konfigurasi yang dihasilkan.
4.	<code>get_moves(state, input, stack, config)</code>	Fungsi untuk mendapatkan semua langkah yang mungkin pada PDA. Input: State saat ini, input saat ini, tumpukan saat ini, konfigurasi saat ini. Output: Array langkah-langkah yang mungkin.
5.	<code>is_found(state, input, stack)</code>	Fungsi untuk menentukan apakah PDA telah menemukan konfigurasi yang diterima. Input: State saat ini, input saat ini, tumpukan saat ini. Output: True jika diterima, False jika tidak.
6.	<code>print_config(config)</code>	Prosedur untuk mencetak konfigurasi PDA. Input: Konfigurasi PDA.
7.	<code>parse_file(filename)</code>	Prosedur untuk membaca file PDA dan menginisialisasi variabel-variabel global. Input: Nama file PDA. Output: True jika berhasil, False jika gagal.
8.	<code>done()</code>	Prosedur untuk mencetak hasil akhir apakah PDA diterima atau tidak.
9.	<code>search_error_location(file_path, error_input)</code>	Fungsi untuk mencari letak kesalahan sintaksis pada file HTML
10.	<code>main()</code>	Fungsi utama program.

d. Antarmuka

1. Program dijalankan melalui baris perintah dengan menyediakan dua argumen: file PDA dan file HTML, dalam hal ini file html berada dalam folder test.

```
python checker.py pda.txt test/test.html
python3 checker.py pda.txt test/test.html
```

2. Program mencetak output apakah HTML diterima atau tidak sesuai dengan aturan PDA.

Accepted or Syntax Error

BAB IV

Uji Kasus

4.1. Contoh 1 (test.html)

```
<html>
  <head>
    <title>Simple Webpage</title>
    <script>
      document.getElementById("demo").innerHTML = "Hello JavaScript!";
    </script>
  </head>
  <body>
    <h1 id="element">The script element</h1>
    <a>Not going anywhere</a><br>
    <a href="https://www.google.co.id/">Might send you somewhere</a>

    <p id="demo"></p>
  </body>
</html>
```

```
PS C:\Users\ASUS\OneDrive\Documents\GitHub\TUBES-TBFO\src> python checker.py pda.txt test/test.html
Accepted
PS C:\Users\ASUS\OneDrive\Documents\GitHub\TUBES-TBFO\src> █
```

Keterangan: File memenuhi aturan sintaksis sehingga diberikan *verdict* “Accepted”

4.2. Contoh 2 (test2.html)

```
<html>
  <head>
    <title>Simple Webpage</title>
  </head>
  <body>
    <!-- Bagian utama web -->
    <h1>Hello, World!</h1>
    <h2>Welcome to my page</h2>
    <hr>
    
```



```

<p>This is a <em>simple</em> webpage.</p>

<!-- Custom element -->
<div id="footer" class="footer"> This is the end of the page </div>
</body>
</html>

```

```

PS C:\Users\ASUS\OneDrive\Documents\GitHub\TUBES-TBFO\src> python checker.py pda.txt test/test2.html
Accepted
PS C:\Users\ASUS\OneDrive\Documents\GitHub\TUBES-TBFO\src> 

```

Keterangan: File memenuhi aturan sintaksis sehingga diberikan *verdict* “Accepted”

4.3. Contoh 3 (test3.html)

```

<html>
<head>
  <title>Simple Webpage</title>
</head>
<body>
  <!-- Bagian utama web -->
  <h1>Hello, World!</h1>
  <h2>Welcome to my page</h2>
  <img alt="Welcome Banner">
  <p>This is a <em>simple</em> webpage.</p>

  <!-- Custom element -->
  <div id="footer" class="footer"> This is the end of the page </div>
</body>
</html>

```

```

● williamglory@Williams-MBP src % python3 checker.py pda.txt test/test3.html
Syntax Error.
Terjadi kesalahan pada line 9:
<img alt="Welcome Banner">
○ williamglory@Williams-MBP src % 

```

Keterangan : Syntax Error karena atribut wajib src pada img tidak ada

4.4. Contoh 4 (test4.html)

```

<html>

```

```

<head>
  <title>Simple Webpage</title>
</head>
<body>
  <h2>HTML Forms</h2>

  <form action="/action_page.php" method="TEMLAK">
    <div id="label">First name:</div>
    <br />
    <input type="text" id="fname" /><br />
    <div id="label">Last name:</div>
    <br />
    <input type="text" id="lname" /><br /><br />
    <button type="submit">Submit</button>
  </form>

  <p>
    If you click the "Submit" button, the form-data will be sent to a
page
    called "/action_page.php".
  </p>
</body>
</html>

```

```

● williamglory@Williams-MBP src % python3 checker.py pda.txt test/test4.html
Syntax Error.
Terjadi kesalahan pada line 8:
<form action="/action_page.php" method="TEMLAK">
○ williamglory@Williams-MBP src % █

```

Keterangan : value pada method tidak terdefinisi yaitu selain POST, GET

BAB V

Sumber Terkait dan Pembagian Kerja

Link Repository : <https://github.com/BocilBlunder/Tugas-Pemrograman-TBFO.git>

Link Diagram State : <https://app.diagrams.net/?src=about#G1DnK3-cbAy4yRasc3L-wuWsjokGGmUHG9>

Pembagian Kerja

Nama	NIM	Jobdesc
Ahmad Mudabbir Arif	13522072	Checker, PDA, Diagram, Testing, Bonus, dan Laporan
Muhammad Neo Cicero Koda	13522108	Checker, PDA, Diagram, Testing, Bonus, dan Laporan
William Glory Henderson	13522113	Checker, PDA, Diagram, Testing, Bonus, dan Laporan

BAB VI

Daftar Pustaka

Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2006). Introduction to Automata Theory, Languages, and Computation. Pearson.

[Introduction of Pushdown Automata - GeeksforGeeks](#)