LAPORAN TUGAS BESAR 1 IF2124

Teori Bahasa Formal dan Otomata

2023/2024

Kelompok 2:

Evelyn Yosiana / 13522083

Fedrianz Dharma / 13522090

Steven Tjhia / 13522103

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

DAFTAR ISI

DAFTAR ISI **2**
DAFTAR GAMBAR **4**
BAB 1 ..... **5**
BAB 2 ..... **6**
2.1. HTML 6
2.2. PDA 8
Gambar 2.1 Ilustrasi PDA 8
Gambar 2.2 Transisi PDA 9
2.3. Hubungan PDA dengan HTML 9
BAB 3 ..... **10**
3.1. PDA 10
3.2. Tokenisasi 30
3.3. Asumsi 31
BAB 4 ..... **32**
3.1. File automata.py 32
3.1.1. def readTxt(filename) 32
3.1.2. def top(stack) 33
3.1.3. def rulesprocess() 33
3.1.4. def process() 33
3.2. File countag.py 33
3.3. File globalvar.py 33
3.4. File parsing.py 33
3.4.1. def editArr(arr) 33
3.4.2. def tanganiTeks(arr, item) 34
3.4.3. handle_starttag(self, tag, attrs) 34
3.4.4. handle_endtag(self, tag) 34
3.4.5. handle_data(self, data) 34
3.4.6. handle_comment(self, data) 34
BAB 5 ..... **35**
5.1. Testcase 1 35
5.2. Testcase 2 36
5.3. Testcase 3 37
5.4. Testcase 4 38
5.5. Testcase 5 39
5.6. Testcase 6 40
5.7. Testcase 7 41
5.8. Testcase 8 42

5.9. Testcase 9....43
5.10. Testcase 10....44
5.11. Testcase 11....45
5.12. Testcase 12....46
5.13. Testcase 13....47
BAB 6....48
6.1. Kesimpulan....48
6.2. Saran....48
BAB 7....49
7.1. Pembagian Tugas....49
LAMPIRAN....50
DAFTAR PUSTAKA....51

DAFTAR GAMBAR

Gambar 2.1 Ilustrasi PDA..... 7
Gambar 2.2 Transisi PDA..... 8
Gambar 5.1.1 Testcase 1.....33
Gambar 5.1.2 Hasil Testcase 1.....33
Gambar 5.2.1 Testcase 2.....34
Gambar 5.2.2 Hasil Testcase 2.....34
Gambar 5.3.1 Testcase 3.....35
Gambar 5.3.2 Hasil Testcase 3.....35
Gambar 5.4.1 Testcase 4.....36
Gambar 5.4.2 Hasil Testcase 4.....36
Gambar 5.5.1 Testcase 5.....37
Gambar 5.5.2 Hasil Testcase 5.....37
Gambar 5.6.1 Testcase 6.....38
Gambar 5.6.2 Hasil Testcase 6.....38
Gambar 5.7.1 Testcase 7.....39
Gambar 5.7.2 Hasil Testcase 7.....39
Gambar 5.8.1 Testcase 8.....40
Gambar 5.8.2 Hasil Testcase 8.....40
Gambar 5.9.1 Testcase 9.....41
Gambar 5.9.2 Hasil Testcase 9.....41
Gambar 5.10.1 Testcase 10.....42
Gambar 5.10.2 Hasil Testcase 10.....42
Gambar 5.11.1 Testcase 11.....43
Gambar 5.11.2 Hasil Testcase 11.....43

BAB 1

Deskripsi Permasalahan

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, <img> 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, implementasikan sebuah program yang dapat memeriksa kebenaran HTML dari segi nama *tag* yang digunakan serta *attribute* yang dimilikinya. Pada tugas pemrograman ini, gunakanlah konsep Pushdown Automata (PDA) dalam mencapai hal tersebut yang diimplementasikan dalam bahasa Python.

BAB 2

Landasan Teori

2.1. HTML

HTML (Hypertext Markup Language) yaitu bahasa makrup standar yang digunakan dalam pembuatan *website.* Dapat dikatakan bahwa HTML merupakan fondasi yang menjadi kerangka dalam pembuatan website. Saat ini, HTML dikelola oleh World Wide Web Consortium (W3C). W3C juga yang bertugas dalam merilis pembaruan HTML.

Ekstensi HTML adalah ".html". Halaman-halaman yang dibangun menggunakan HTML terdiri atas serangkaian tag. Sebagian besar tag terdiri atas tag pembuka dan tag penutup, misalnya tag ditutup dengan tag . Namun, terdapat beberapa tag yang tidak memiliki penutup tag, misalnya
 untuk membuat baris baru, <img> untuk memasukkan gambar dalam web, dan <link> untuk menghubungkan halaman web dengan halaman yang lain.

HTML memiliki banyak tag dengan fungsinya masing-masing. Beberapa tag html yang sering digunakan antara lain:

- **html**, merepresentasikan akar dari file HTML;
- **head**, merupakan kontainer untuk metadata;
- **body**, berisi seluruh konten web yang akan dibuat;
- **title**, merupakan judul halaman;
- **link**, memungkinkan halaman web terhubung dengan halaman lainnya;
- **script**, untuk menghubungkan html dengan javascript (*backend*);
- **h1**, **h2**, **h3**, **h4**, **h5**, **h6**, mendefinisikan heading;
- **p**, mendefinisikan paragraf;
- **br**, untuk berganti ke baris baru;
- **em**, untuk menampilkan kalimat dengan *italic*;
- **b**, untuk menampilkan kalimat dengan bold;
- **abbr**, mendefinisikan akronim;

- **strong**, mendefinisikan kata yang penting dalam bold;
- **small**, untuk menampilkan kalimat dengan ukuran yang lebih kecil;
- **hr**, untuk menampilkan garis batasan pada halaman;
- **div**: untuk mendefinisikan sebuah bagian dalam halaman;
- **a**: untuk mendefinisikan tautan untuk terhubung dengan halaman web lainnya;
- **img**: untuk menampilkan gambar pada halaman web;
- **button**: untuk menambahkan tombol pada halaman web;
- **form**: untuk membuat form yang dapat diisi oleh pengguna;
- **input**: untuk membuat tabel input yang dapat diisi dengan sesuatu oleh pengguna;
- **table**: untuk membuat dan menampilkan tabel pada halaman web;
- **tr**: mendefinisikan baris dalam tabel HTML;
- **td**: mendefinisikan *cell* dalam tabel HTML;
- **th**: mendefinisikan *header cell* dalam tabel HTML.

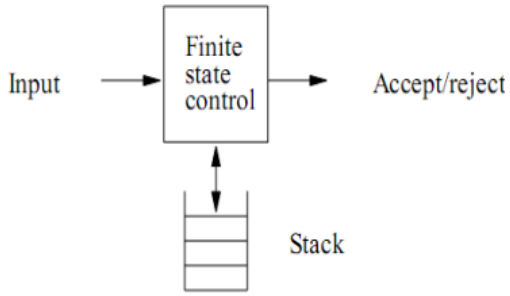
Berikut adalah struktur dasar HTML.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>

</body>
</html>
```

2.2. PDA

Pushdown automata (PDA) merupakan salah satu bentuk finite automata yang memiliki memori tambahan berupa *stack*. Tambahan stack ini memungkinkan PDA untuk mengenali *context free language*. PDA dapat diilustrasikan sebagai berikut.



Gambar 2.1 Ilustrasi PDA

PDA dapat didefinisikan dengan 7 *tuples* sebagai berikut:

Q: himpunan state;

$\sum$: himpunan masukan simbol;

Γ: himpunan stack;

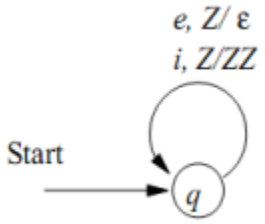
$q0$: state awal;

δ: fungsi transisi;

$Z0$: simbol stack awal;

F: himpunan final state.

Notasi transisi dari PDA dapat digambarkan sebagai berikut.



Gambar 2.2 Transisi PDA

Berdasarkan gambar di atas, e dan i merupakan simbol masukan, q merupakan state, Z/ε dan Z/ZZ merupakan keadaan *stack*. Jika simbol masukan e, maka *stack* akan melakukan *pop* atau mengeluarkan Z jika Z merupakan elemen puncak (*top*) dari *stack*. Jika simbol masukan i, maka *stack* akan melakukan *push* atau memasukkan Z sebagai elemen puncak dari *stack* jika Z merupakan elemen puncak *stack* sebelumnya. Jika simbol masukan e dan i maka state sekarang akan berpindah ke dirinya sendiri (q). Jika masukan selain e dan i maka masukan tidak valid.

2.3. Hubungan PDA dengan HTML

Dalam struktur HTML, tidak jarang ditemui struktur bersarang (*nested*). Untuk memproses struktur seperti ini, *pushdown automata* (PDA) merupakan salah satu mesin otomata yang cocok. Hal tersebut dikarenakan PDA memiliki stack yang dapat melacak state-state yang dalam konteks program kali ini yaitu tag-tag HTML terutama tag HTML yang memiliki tag buka dan tag tutup. Dengan PDA, kita dapat melihat apakah tag buka HTML telah tertutup dengan baik atau belum dengan bantuan stack.

BAB 3

Penjelasan PDA

$P = (Q, \sum, \Gamma, q0, \delta, Z0, F)$

3.1. PDA

Berikut PDA yang kami buat.

Q = {q0, q1, q2, q3, q4, q5, q6, q7, q8, q9, q10, q11, q12, q13, q14, q15, q16, q17, q18, q19, q20, q21, q22, q23, q24, q25, q26, q27, q28, q29, q30, q31, q32, q33, q34, q35, q36, q37, q38, q39, q40, q41, q42, q43, q44, q45, q46, q47, q48, q49, q50, q51, q52, q53, q54, q55, q56, q57, q58, q59, q60, q61, q62, q63, q64, q65, q66, q67, q68, q69, q70, q71, q72, q73, q74, q75, q76, q77, q78, q79, q80, q81, q82, q83, q84, q85, q86, q87, q88, q89, q90, q91, q92, q93, q94, q95, q96, q97, q98, qf}.

∑ = {<h2>, <h3>, <h4>, <h5>, <h6>, , , , <abbr>, , <small>, , <th>, <td>, <tr>, <table>, <img>,
, <hr>, <a>, <button>, <link>, <form>, <input>, Rel, href, src, alt, type, action, method, </html>, </head>, </body>, </title>, </script>, </h1>, </h2>, </h3>, </h4>, </h5>, </h6>, , , , </abbr>, , </small>, , </th>, </td>, </tr>, >, ", get, />, post, submit, reset, button, </button>, </form>, , </script>, %, eps, class, id, style, text , password, email, number, checkbox, input, =, rel, </table>}.

Γ = {A, B, C, D, E, F, G, H, I, J, K, L, M, N, P, Q, R, S, T, U, V, W, X, Y, Z, !, @, #, $, &}.

q0 = q0.

Fungsi transisi (δ):

δ(q0, <html>, Z) = (q1, AZ)

δ(q1, <head>, A) = (q80, BA)

δ(q1, *class=, A) = (q81, A)

δ(q1, *id=, A) = (q81, A)

δ(q1, *style, A) = (q81, A)

δ(q81, " A) = (q82, A)

δ(q82, % I) = (q82, I)

δ(q82, " I) = (q1, eps)

δ(q80, *class=, B) = (q78, B)

δ(q80, *id=, B) = (q78, B)

δ(q80, *style, B) = (q78, B)

δ(q78, " B) = (q79, IB)

δ(q79, % I) = (q79, I)

δ(q79, " I) = (q80, eps)

δ(q80, <title>, B) = (q45, CB)

δ(q80, <link>, B) = (q6, B)

δ(q80, <script>, B) = (q4, EB)

δ(q80, </head>, B) = (q44, eps)

δ(q2, <title>, B) = (q45, CB)

δ(q3, <title>, B) = (q45, CB)

δ(q24, <title>, B) = (q45, CB)

δ(q45, % C) = (q46, C)

δ(q45, </title>, C) = (q2, eps)

δ(q45, *class=, C) = (q83, C)

δ(q45, *id=, C) = (q83, C)

δ(q45, *style, C) = (q83, C)

δ(q83, " C) = (q84, IC)

δ(q84, " I) = (q45, eps)

δ(q84, % I) = (q84, I)

δ(q46, </title>, C) = (q2, eps)

δ(q2, <link>, B) = (q6, B)

δ(q24, <link>, B) = (q6, B)

δ(q3, <link>, B) = (q6, B)

δ(q6, rel=, B) = (q22, B)

δ(q6, href=, B) = (q85, B)

δ(q6, *class=, B) = (q85, B)

δ(q6, *id=, B) = (q85, B)

δ(q6, *style, B) = (q85, B)

δ(q85, " B) = (q86, IB)

δ(q86, " I) = (q6, eps)

δ(q86, % I) = (q86, I)

δ(q22, " B) = (q23, IB)

δ(q23, % I) = (q23, I)

δ(q23, " I) = (q24, eps)

δ(q24, % B) = (q3, B)

δ(q24, href=, B) = (q25, B)

δ(q24, *class=, B) = (q25, B)

δ(q24, *id=, B) = (q25, B)

δ(q24, *style, B) = (q25, B)

δ(q25, " B) = (q26, IB)

δ(q26, % I) = (q26, I)

δ(q26, " I) = (q24, eps)

δ(q2, <script>, B) = (q4, EB)

δ(q24, <script>, B) = (q4, EB)

δ(q3, <script>, B) = (q4, EB)

δ(q4, % E) = (q5, E)

δ(q4, </script>, E) = (q2, eps)

δ(q4, src=, E) = (q32, E)

δ(q4, *class=, E) = (q32, E)

δ(q4, *id=, E) = (q32, E)

δ(q4, *style, E) = (q32, E)

δ(q32, " E) = (q48, IE)

δ(q48, % I) = (q48, I)

δ(q48, " I) = (q4, eps)

δ(q5, </script>, E) = (q2, eps)

δ(q2, </head>, B) = (q44, eps)

δ(q3, </head>, B) = (q44, eps)

δ(q24, </head>, B) = (q44, eps)

δ(q44, <body>, A) = (q47, FA)

δ(q7, , F) = (q96, PF)

δ(q7, , P) = (q96, PP)

δ(q7, , T) = (q96, PT)

δ(q7, , P) = (q7, eps)

δ(q7, </form>, T) = (q7, eps)

δ(q7, <h1>, F) = (q8, GF)

δ(q7, <h1>, P) = (q8, GP)

δ(q7, <h1>, T) = (q8, GT)

δ(q7, <h2>, F) = (q8, !F)

δ(q7, <h2>, P) = (q8, !P)

δ(q7, <h2>, T) = (q8, !T)

δ(q7, <h3>, F) = (q8, @F)

δ(q7, <h3>, P) = (q8, @P)

δ(q7, <h3>, T) = (q8, @T)

δ(q7, <h4>, F) = (q8, #F)

δ(q7, <h4>, P) = (q8, #P)

δ(q7, <h4>, T) = (q8, #T)

δ(q7, <h5>, F) = (q8, $F)

δ(q7, <h5>, P) = (q8, $P)

δ(q7, <h5>, T) = (q8, $T)

δ(q7, <h6>, F) = (q8, &F)

δ(q7, <h6>, P) = (q8, &P)

δ(q7, <h6>, T) = (q8, &T)

δ(q8, % G) = (q9, G)

δ(q8, % !) = (q9, !)

δ(q8, % @) = (q9, @)

δ(q8, % #) = (q9, #)

δ(q8, % $) = (q9, $)

δ(q8, % &) = (q9, &)

δ(q8, *class= G) = (q87, G)

δ(q8, *id= G) = (q87, G)

δ(q8, *style G) = (q87, G)

δ(q8, *class= !) = (q87, !)

δ(q8, *id= !) = (q87, !)

δ(q8, *style !) = (q87, !)

δ(q8, *class= @) = (q87, @)

δ(q8, *id= @) = (q87, @)

δ(q8, *style @) = (q87, @)

δ(q8, *class= #) = (q87, #)

δ(q8, *id= #) = (q87, #)

δ(q8, *style #) = (q87, #)

δ(q8, *class= $) = (q87, $)

δ(q8, *id= $) = (q87, $)

δ(q8, *style $) = (q87, $)

δ(q8, *class= &) = (q87, &)

δ(q8, *id= &) = (q87, &)

δ(q8, *style &) = (q87, &)

δ(q87, " G) = (q88, IG)

δ(q87, " !) = (q88, I!)

δ(q87, " @) = (q88, I@)

δ(q87, " #) = (q88, I#)

δ(q87, " $) = (q88, I$)

δ(q87, " &) = (q88, I&)

δ(q88, % I) = (q88, I)

δ(q88, " I) = (q8, eps)

δ(q9, </h1>, G) = (q7, eps)

δ(q9, </h2>, !) = (q7, eps)

δ(q9, </h3>, @) = (q7, eps)

δ(q9, </h4>, #) = (q7, eps)

δ(q9, </h5>, $) = (q7, eps)

δ(q9, </h6>, &) = (q7, eps)

δ(q8, </h1>, G) = (q7, eps)

δ(q8, </h2>, !) = (q7, eps)

δ(q8, </h3>, @) = (q7, eps)

δ(q8, </h4>, #) = (q7, eps)

δ(q8, </h5>, $) = (q7, eps)

δ(q8, </h6>, &) = (q7, eps)

δ(q7, <p>, F) = (q11, HF)

δ(q7, <p>, P) = (q11, HP)

δ(q7, <p>, T) = (q11, HT)

δ(q11, % H) = (q10, H)

δ(q11, , H) = (q12, JH)

δ(q11, , H) = (q12, KH)

δ(q11, <abbr>, H) = (q12, LH)

δ(q11, , H) = (q12, MH)

δ(q11, <small>, H) = (q12, NH)

δ(q10, , H) = (q12, JH)

δ(q10, , H) = (q12, KH)

δ(q10, <abbr>, H) = (q12, LH)

δ(q10, , H) = (q12, MH)

δ(q10, <small>, H) = (q12, NH)

δ(q10, % H) = (q10, H)

δ(q12, % J) = (q13, J)

δ(q14, % K) = (q15, K)

δ(q16, % L) = (q17, L)

δ(q18, % M) = (q19, M)

δ(q20, % N) = (q21, N)

δ(q13, , J) = (q10, eps)

δ(q15, , K) = (q10, eps)

δ(q15, </abbr>, L) = (q10, eps)

δ(q15, , M) = (q10, eps)

δ(q15, </small>, N) = (q10, eps)

δ(q10, </p>, H) = (q7, eps)

δ(q7,
, F) = (q96, F)

δ(q7,
, P) = (q96, P)

δ(q7,
, T) = (q96, T)

δ(q7, <hr>, F) = (q96, F)

δ(q7, <hr>, P) = (q96, P)

δ(q7, <hr>, T) = (q96, T)

δ(q7, % T) = (q7, T)

δ(q7, % P) = (q7, P)

δ(q7, <img>, F) = (q53, F)

δ(q7, <img>, P) = (q53, P)

δ(q7, <img>, T) = (q53, T)

δ(q53, src=, F) = (q54, F)

δ(q53, src=, P) = (q54, P)

δ(q53, src=, T) = (q54, T)

δ(q53, " F) = (q55, IF)

δ(q53, " P) = (q55, IP)

δ(q53, " T) = (q55, IT)

δ(q55, % I) = (q55, I)

δ(q55, " I) = (q56, eps)

δ(q56, alt=, F) = (q57, F)

δ(q56, alt=, P) = (q57, P)

δ(q56, alt=, T) = (q57, T)

δ(q56, *class=, F) = (q57, F)

δ(q56, *class=, P) = (q57, P)

δ(q56, *class=, T) = (q57, T)

δ(q56, *id=, F) = (q57, F)

δ(q56, *id=, P) = (q57, P)

δ(q56, *id=, T) = (q57, T)

δ(q56, *style, F) = (q57, F)

δ(q56, *style, P) = (q57, P)

δ(q56, *style, T) = (q57, T)

δ(q53, alt=, F) = (q94, F)

δ(q53, alt=, P) = (q94, P)

δ(q53, alt=, T) = (q94, T)

δ(q53, *class=, F) = (q94, F)

δ(q53, *class=, P) = (q94, P)

δ(q53, *class=, T) = (q94, T)

δ(q53, *id=, F) = (q94, F)

δ(q53, *id=, P) = (q94, P)

δ(q53, *id=, T) = (q94, T)

δ(q53, *style, F) = (q94, F)

δ(q53, *style, P) = (q94, P)

δ(q53, *style, T) = (q94, T)

δ(q57, " F) = (q58, IF)

δ(q57, " P) = (q58, IP)

δ(q57, " T) = (q58, IT)

δ(q94, " F) = (q95, IF)

δ(q94, " P) = (q95, IP)

δ(q94, " T) = (q95, IT)

δ(q95, % I) = (q95, I)

δ(q58, % I) = (q58, I)

δ(q58, " I) = (q56, eps)

δ(q95, " I) = (q53, eps)

δ(q7, <a>, F) = (q27, QF)

δ(q7, <a>, P) = (q27, QP)

δ(q7, <a>, T) = (q27, QT)

δ(q27, , Q) = (q7, eps)

δ(q27, % Q) = (q28, Q)

δ(q27, href=, Q) = (q50, Q)

δ(q27, *class=, Q) = (q50, Q)

δ(q27, *id=, Q) = (q50, Q)

δ(q27, *style, Q) = (q50, Q)

δ(q50, " Q) = (q51, IQ)

δ(q51, % I) = (q51, I)

δ(q51, " I) = (q52, eps)

δ(q52, % Q) = (q28, Q)

δ(q52, , Q) = (q7, eps)

δ(q28, , Q) = (q7, eps)

δ(q7, <button>, F) = (q29, SF)

δ(q7, <button>, P) = (q29, SP)

δ(q7, <button>, T) = (q29, ST)

δ(q29, type=, S) = (q59, S)

δ(q29, *class=, S) = (q59, S)

δ(q29, *id=, S) = (q59, S)

δ(q29, *style, S) = (q59, S)

δ(q29, </button>, S) = (q7, eps)

δ(q29, % S) = (q30, S)

δ(q59, " S) = (q60, IS)

δ(q60, submit, I) = (q61, I)

δ(q60, reset, I) = (q62, I)

δ(q60, button, I) = (q63, I)

δ(q61, " I) = (q29, eps)

δ(q62, " I) = (q29, eps)

δ(q63, " I) = (q29, eps)

δ(q30, </button>, S) = (q7, eps)

δ(q7, <table>, F) = (q33, UF)

δ(q7, <table>, P) = (q33, UP)

δ(q7, <table>, T) = (q33, UT)

δ(q33, </table>, U) = (q7, eps)

δ(q33, <tr>, U) = (q34, VU)

δ(q34, <tr>, V) = (q38, eps)

δ(q34, <td>, V) = (q40, XV)

δ(q34, <th>, V) = (q35, WV)

δ(q35, % W) = (q36, W)

δ(q35, </th>, W) = (q37, eps)

δ(q36, </th>, W) = (q37, eps)

δ(q37, </tr>, V) = (q38, eps)

δ(q37, <th>, V) = (q35, WV)

δ(q38, </table>, U) = (q7, eps)

δ(q38, <tr>, U) = (q39, VU)

δ(q39, <td>, V) = (q40, XV)

δ(q39, </tr>, V) = (q38, eps)

δ(q40, </td>, X) = (q42, eps)

δ(q40, % X) = (q41, X)

δ(q41, </td>, X) = (q42, eps)

δ(q42, <td>, V) = (q40, XV)

δ(q7, <form>, F) = (q31, TF)

δ(q7, <form>, P) = (q31, TP)

δ(q31, action=, T) = (q64, T)

δ(q31, *class=, T) = (q64, T)

δ(q31, *id=, T) = (q64, T)

δ(q31, *style, T) = (q64, T)

δ(q64, " T) = (q65, IT)

δ(q65, % I) = (q65, I)

δ(q65, " I) = (q31, eps)

δ(q31, method=, T) = (q66, T)

δ(q66, " T) = (q67, IT)

δ(q67, get, I) = (q68, I)

δ(q67, post, I) = (q69, I)

δ(q68, " I) = (q31, eps)

δ(q69, " I) = (q31, eps)

δ(q31, , T) = (q96, PT)

δ(q31, </form>, T) = (q7, eps)

δ(q31, <h1>, T) = (q8, GT)

δ(q31, <h2>, T) = (q8, !T)

δ(q31, <h3>, T) = (q8, @T)

δ(q31, <h4>, T) = (q8, #T)

δ(q31, <h5>, T) = (q8, $T)

δ(q31, <h6>, T) = (q8, &T)

δ(q31, <p>, T) = (q11, HT)

δ(q31, <input>, T) = (q70, T)

δ(q31,
, T) = (q96, T)

δ(q31, <hr>, T) = (q96, T)

δ(q31, %, T) = (q96, T)

δ(q31, <img>, T) = (q53, T)

δ(q31, <a>, T) = (q27, QT)

δ(q31, <button>, T) = (q29, ST)

δ(q31, <table>, T) = (q33, UT)

δ(q7, </body>, F) = (q43, eps)

δ(q43, </html>, A) = (qf, eps)

δ(q47, , F) = (q96, PF)

δ(q47, <h1>, F) = (q8, GF)

δ(q47, <h2>, F) = (q8, !F)

δ(q47, <h3>, F) = (q8, @F)

δ(q47, <h4>, F) = (q8, #F)

δ(q47, <h5>, F) = (q8, $F)

δ(q47, <h6>, F) = (q8, &F)

δ(q47, <p>, F) = (q11, HF)

δ(q47,
, F) = (q96, F)

δ(q47, <hr>, F) = (q96, F)

δ(q47, <img>, F) = (q53, F)

δ(q47, <a>, F) = (q27, QF)

δ(q47, <button>, F) = (q29, SF)

δ(q47, <table>, F) = (q33, UF)

δ(q47, <form>, F) = (q31, TF)

δ(q47, *class=, F) = (q49, F)

δ(q47, *id=, F) = (q49, F)

δ(q47, *style, F) = (q49, F)

δ(q47, </body>, F) = (q43, eps)

δ(q49, " F) = (q89, IF)

δ(q89, % I) = (q89, I)

δ(q89, " I) = (q47, eps)

δ(q39, *class=, V) = (q92, V)

δ(q39, *id=, V) = (q92, V)

δ(q39, *style, V) = (q92, V)

δ(q11, *class=, H) = (q90, H)

δ(q11, *id=, H) = (q90, H)

δ(q11, *style, H) = (q90, H)

δ(q12, *class=, J) = (q90, J)

δ(q12, *id=, J) = (q90, J)

δ(q12, *style, J) = (q90, J)

δ(q13, *class=, K) = (q90, K)

δ(q13, *id=, K) = (q90, K)

δ(q13, *style, K) = (q90, K)

δ(q16, *class=, L) = (q90, L)

δ(q16, *id=, L) = (q90, L)

δ(q16, *style, L) = (q90, L)

δ(q18, *class=, M) = (q90, M)

δ(q18, *id=, M) = (q90, M)

δ(q18, *style, M) = (q90, M)

δ(q20, *class=, N) = (q90, N)

δ(q20, *id=, N) = (q90, N)

δ(q20, *style, N) = (q90, N)

δ(q33, *class=, U) = (q90, U)

δ(q33, *id=, U) = (q90, U)

δ(q33, *style, U) = (q90, U)

δ(q34, *class=, V) = (q90, V)

δ(q34, *id=, V) = (q90, V)

δ(q34, *style, V) = (q90, V)

δ(q35, *class=, W) = (q90, W)

δ(q35, *id=, W) = (q90, W)

δ(q35, *style, W) = (q90, W)

δ(q40, *class=, X) = (q90, X)

δ(q40, *id=, X) = (q90, X)

δ(q40, *style, X) = (q90, X)

δ(q96, *class=, F) = (q90, F)

δ(q96, *id=, F) = (q90, F)

δ(q96, *style, F) = (q90, F)

δ(q96, *class=, P) = (q90, P)

δ(q96, *id=, P) = (q90, P)

δ(q96, *style, P) = (q90, P)

δ(q96, *class=, T) = (q90, T)

δ(q96, *id=, T) = (q90, T)

δ(q96, *style, T) = (q90, T)

δ(q92, " V) = (q93, IV)

δ(q90, " H) = (q91, IH)

δ(q90, " J) = (q91, IJ)

δ(q90, " K) = (q91, IK)

δ(q90, " L) = (q91, IL)

δ(q90, " M) = (q91, IM)

δ(q90, " N) = (q91, IN)

δ(q90, " U) = (q91, IU)

δ(q90, " V) = (q91, IV)

δ(q90, " W) = (q91, IW)

δ(q90, " X) = (q91, IX)

δ(q90, " F) = (q91, IF)

δ(q90, " P) = (q91, IP)

δ(q90, " T) = (q91, IT)

δ(q93, % I) = (q93, I)

δ(q91, % I) = (q91, I)

δ(q91, " I) = (q11, eps)

δ(q93, " I) = (q39, eps)

δ(q91, " I) = (q12, eps)

δ(q91, " I) = (q14, eps)

δ(q91, " I) = (q16, eps)

δ(q91, " I) = (q18, eps)

δ(q91, " I) = (q20, eps)

δ(q91, " I) = (q33, eps)

δ(q91, " I) = (q34, eps)

δ(q91, " I) = (q35, eps)

δ(q91, " I) = (q40, eps)

δ(q91, " I) = (q96, eps)

δ(q70, type=, F) = (q71, F)

δ(q70, type=, P) = (q71, P)

δ(q70, type=, T) = (q71, T)

δ(q70, *class=, F) = (q97, F)

δ(q70, *class=, P) = (q97, P)

δ(q70, *class=, T) = (q97, T)

δ(q70, *id=, F) = (q97, F)

δ(q70, *id=, P) = (q97, P)

δ(q70, *id=, T) = (q97, T)

δ(q70, *style=, F) = (q97, F)

δ(q70, *style=, P) = (q97, P)

δ(q70, *style=, T) = (q97, T)

δ(q97, " F) = (q98, IF)

δ(q97, " P) = (q98, IP)

δ(q98, % I) = (q98, I)

δ(q98, " I) = (q70, eps)

δ(q97, " T) = (q98, IT)

δ(q70, type=, F) = (q71, F)

δ(q70, type=, P) = (q71, P)

δ(q70, type=, T) = (q71, T)

δ(q71, " F) = (q72, IF)

δ(q71, " P) = (q72, IP)

δ(q71, " T) = (q72, IT)

δ(q72, text, I) = (q73, I)

δ(q72, password, I) = (q74, I)

δ(q72, email, I) = (q75, I)

δ(q72, number, I) = (q76, I)

δ(q72, checkbox, I) = (q77, I)

δ(q77, " I) = (q70, eps)

δ(q73, " I) = (q70, eps)

δ(q74, " I) = (q70, eps)

δ(q75, " I) = (q70, eps)

δ(q76, " I) = (q70, eps)

δ(q70, , F) = (q96, PF)

δ(q70, , P) = (q96, PP)

δ(q70, , T) = (q96, PT)

δ(q70,
, F) = (q96, F)

δ(q70,
, P) = (q96, P)

δ(q70,
, T) = (q96, T)

δ(q70, <hr>, F) = (q96, F)

δ(q70, <hr>, P) = (q96, P)

δ(q70, <hr>, T) = (q96, T)

δ(q70, </body>, F) = (q43, eps)

δ(q70, , P) = (q7, eps)

δ(q70, </form>, T) = (q7, eps)

δ(q70, <h1>, F) = (q8, GF)

δ(q70, <h1>, P) = (q8, GP)

δ(q70, <h1>, T) = (q8, GT)

δ(q70, <h2>, F) = (q8, !F)

δ(q70, <h2>, P) = (q8, !P)

δ(q70, <h2>, T) = (q8, !T)

δ(q70, <h3>, F) = (q8, @F)

δ(q70, <h3>, P) = (q8, @P)

δ(q70, <h3>, T) = (q8, @T)

δ(q70, <h4>, F) = (q8, #F)

δ(q70, <h4>, P) = (q8, #P)

δ(q70, <h4>, T) = (q8, #T)

δ(q70, <h5>, F) = (q8, $F)

δ(q70, <h5>, P) = (q8, $P)

δ(q70, <h5>, T) = (q8, $T)

δ(q70, <h6>, F) = (q8, &F)

δ(q70, <h6>, P) = (q8, &P)

δ(q70, <h6>, T) = (q8, &T)

δ(q70, <p>, F) = (q11, HF)

δ(q70, <p>, P) = (q11, HP)

δ(q70, <p>, T) = (q11, HT)

δ(q70, <input>, F) = (q70, F)

δ(q70, <input>, P) = (q70, P)

δ(q70, <input>, T) = (q70, T)

δ(q70, %, T) = (q7, T)

δ(q70, %, P) = (q7, P)

δ(q70, <img>, F) = (q53, F)

δ(q70, <img>, P) = (q53, P)

δ(q70, <img>, T) = (q53, T)

δ(q70, <a>, F) = (q27, QF)

δ(q70, <a>, P) = (q27, QP)

δ(q70, <a>, T) = (q27, QT)

δ(q70, <button>, F) = (q29, SF)

δ(q70, <button>, P) = (q29, SP)

δ(q70, <button>, T) = (q29, ST)

δ(q70, <table>, F) = (q33, UF)

δ(q70, <table>, P) = (q33, UP)

δ(q70, <table>, T) = (q33, UT)

δ(q70, <form>, F) = (q31, TF)

δ(q70, <form>, P) = (q31, TP)

δ(q96, , F) = (q96, PF)

δ(q96, , P) = (q96, PP)

δ(q96, , T) = (q96, PT)

δ(q96,
, F) = (q96, F)

δ(q96,
, P) = (q96, P)

δ(q96,
, T) = (q96, T)

δ(q96, <hr>, F) = (q96, F)

δ(q96, <hr>, P) = (q96, P)

δ(q96, <hr>, T) = (q96, T)

δ(q96, </body>, F) = (q43, eps)

δ(q96, , P) = (q7, eps)

δ(q96, </form>, T) = (q7, eps)

δ(q96, <h1>, F) = (q8, GF)

δ(q96, <h1>, P) = (q8, GP)

δ(q96, <h1>, T) = (q8, GT)

δ(q96, <h2>, F) = (q8, !F)

δ(q96, <h2>, P) = (q8, !P)

δ(q96, <h2>, T) = (q8, !T)

δ(q96, <h3>, F) = (q8, @F)

δ(q96, <h3>, P) = (q8, @P)

δ(q96, <h3>, T) = (q8, @T)

δ(q96, <h4>, F) = (q8, #F)

δ(q96, <h4>, P) = (q8, #P)

δ(q96, <h4>, T) = (q8, #T)

δ(q96, <h5>, F) = (q8, $F)

δ(q96, <h5>, P) = (q8, $P)

δ(q96, <h5>, T) = (q8, $T)

δ(q96, <h6>, F) = (q8, &F)

δ(q96, <h6>, P) = (q8, &P)

δ(q96, <h6>, T) = (q8, &T)

δ(q96, <p>, F) = (q11, HF)

δ(q96, <p>, P) = (q11, HP)

δ(q96, <p>, T) = (q11, HT)

δ(q96, <input>, F) = (q70, F)

δ(q96, <input>, P) = (q70, P)

δ(q96, <input>, T) = (q70, T)

δ(q96, %, T) = (q7, T)

δ(q96, %, P) = (q7, P)

δ(q96, <img>, F) = (q53, F)

δ(q96, <img>, P) = (q53, P)

δ(q96, <img>, T) = (q53, T)

δ(q96, <a>, F) = (q27, QF)

δ(q96, <a>, P) = (q27, QP)

δ(q96, <a>, T) = (q27, QT)

δ(q96, <button>, F) = (q29, SF)

δ(q96, <button>, P) = (q29, SP)

δ(q96, <button>, T) = (q29, ST)

δ(q96, <table>, F) = (q33, UF)

δ(q96, <table>, P) = (q33, UP)

δ(q96, <table>, T) = (q33, UT)

δ(q96, <form>, F) = (q31, TF)

δ(q96, <form>, P) = (q31, TP)

δ(q56, , F) = (q96, PF)

δ(q56, , P) = (q96, PP)

δ(q56, , T) = (q96, PT)

δ(q56,
, F) = (q96, F)

δ(q56,
, P) = (q96, P)

δ(q56,
, T) = (q96, T)

δ(q56, <hr>, F) = (q96, F)

δ(q56, <hr>, P) = (q96, P)

δ(q56, <hr>, T) = (q96, T)

δ(q56, </body>, F) = (q43, eps)

δ(q56, , P) = (q7, eps)

δ(q56, </form>, T) = (q7, eps)

δ(q56, <h1>, F) = (q8, GF)

δ(q56, <h1>, P) = (q8, GP)

δ(q56, <h1>, T) = (q8, GT)

δ(q56, <h2>, F) = (q8, !F)

δ(q56, <h2>, P) = (q8, !P)

δ(q56, <h2>, T) = (q8, !T)

δ(q56, <h3>, F) = (q8, @F)

δ(q56, <h3>, P) = (q8, @P)

δ(q56, <h3>, T) = (q8, @T)

δ(q56, <h4>, F) = (q8, #F)

δ(q56, <h4>, P) = (q8, #P)

δ(q56, <h4>, T) = (q8, #T)

δ(q56, <h5>, F) = (q8, $F)

δ(q56, <h5>, P) = (q8, $P)

δ(q56, <h5>, T) = (q8, $T)

δ(q56, <h6>, F) = (q8, &F)

δ(q56, <h6>, P) = (q8, &P)

δ(q56, <h6>, T) = (q8, &T)

δ(q56, <p>, F) = (q11, HF)

δ(q56, <p>, P) = (q11, HP)

δ(q56, <p>, T) = (q11, HT)

δ(q56, <input>, F) = (q70, F)

δ(q56, <input>, P) = (q70, P)

δ(q56, <input>, T) = (q70, T)

δ(q7, <input>, F) = (q70, F)

δ(q7, <input>, P) = (q70, P)

δ(q7, <input>, T) = (q70, T)

δ(q56, %, T) = (q7, T)

δ(q56, %, P) = (q7, P)

δ(q56, <img>, F) = (q53, F)

δ(q56, <img>, P) = (q53, P)

δ(q56, <img>, T) = (q53, T)

δ(q56, <a>, F) = (q27, QF)

δ(q56, <a>, P) = (q27, QP)

δ(q56, <a>, T) = (q27, QT)

δ(q56, <button>, F) = (q29, SF)

δ(q56, <button>, P) = (q29, SP)

δ(q56, <button>, T) = (q29, ST)

δ(q56, <table>, F) = (q33, UF)

δ(q56, <table>, P) = (q33, UP)

δ(q56, <table>, T) = (q33, UT)

δ(q56, <form>, F) = (q31, TF)

δ(q56, <form>, P) = (q31, TP)

δ(q54, '', F) = (q55, IF)

δ(q54, '', P) = (q55, IP)

δ(q54, '', T) = (q55, IT)

δ(q55, ", I) = (q56, eps)

δ(q67, get, I) = (q68, I)

δ(q67, post, I) = (q69, I)

δ(q34, </tr>, V) = (q38, eps)

*Z*0: Z.

F: {qf}.

3.2. Tokenisasi

Untuk mempermudah pemrosesan, dilakukan tokenisasi pada beberapa simbol, antara lain:

Simbol	Hasil tokenisasi
<Seluruh komen>	comment
id	*id
style	*style
class	*class
rel	rel=
href	href=
src	src=
alt	alt=
type	type=
action	action=
method	method=
<Seluruh random teks>	%

Untuk desain PDA dapat dilihat pada link berikut atau dapat dilihat pada lampiran.

3.3. Asumsi

Karena adanya keterbatasan-keterbatasan tertentu, maka pada program ini, kami memiliki asumsi sebagai berikut.

- <table> tanpa isi diterima;
- <tr> tanpa isi diterima;
- <tr> di dalam <tr> diterima;
- <td> tanpa isi diterima, namun harus berada dalam <tr>;
- <table> hanya boleh berisi <tr>;
- <tr> berisi <th> atau <td> diterima;
- <table> hanya dapat berisi <tr>;
- <tr> berisi <th> atau <td> diterima;
- Jika ada <th>, pasti terletak pada <tr> pertama;
- <th> hanya boleh berada dalam <td> pertama;
- <td> pertama tanpa <th> diterima.
- <th> dan <td> hanya bisa berisi string, tidak dapat berisi tag lain;
- Value berisi spasi tidak diterima;
- <form> dalam <form> tidak diterima;
- Tag untuk string formatting yang diterima hanya: <b>, <abbr>, , <small>, dan .

BAB 4

Implementasi

Implementasi dari program kami terdiri dari empat file python utama, dua file html, dan satu file txt. Berikut directory dari program kami:

src
| — automata.py
| — counttag.py
| — globalvar.py
| — index.html
| — indexlower.html
| — parsing.py
| — pda.txt

Untuk mengecek kevalidan dari suatu struktur HTML, pengguna dapat memasukkannya ke dalam file index.html yang kemudian seluruh isinya akan diubah menjadi huruf kecil kemudian dimasukkan ke dalam file indexlower.html. Program akan membaca file dari indexlower.html. File pda.txt berisi *rules* dari PDA program kami. Empat file python akan dijelaskan di bawah ini.

3.1. File automata.py

Dalam file automata.py, terdapat empat fungsi yaitu readTxt, top, rulesprocess, dan process.

3.1.1. def readTxt(filename)

Fungsi readTxt menerima parameter berupa path ke file tertentu (filename). Fungsi ini berguna membaca file txt kemudian memasukkan isinya ke dalam sebuah global variable.

3.1.2. def top(stack)

Fungsi top berfungsi untuk mendapatkan nilai elemen puncak dari stack.

3.1.3. def rulesprocess()

Fungsi rulesprocess digunakan untuk memproses fungsi trasnisisi dari file pda.txt menjadi *nested dictionary.*

3.1.4. def process()

Fungsi process berguna untuk mengecek file HTML dari pengguna dengan fungsi transisi yang telah ada sehingga didapatkan hasil file tersebut valid atau tidak menggunakan prinsip PDA.

3.2. File countag.py

Dalam file automata.py, terdapat array linier berukuran 41 untuk menyimpan berapa kali suatu tag telah melewati PDA.

3.3. File globalvar.py

File globalvar.py berisi *tuples* PDA yang digunakan dalam program ini.

3.4. File parsing.py

Dalam file parsing.py, terdapat beberapa fungsi untuk membaca file dari file html, melakukan *parsing* terhadap tag-tag di dalamnya, serta modifikasi dan manipulasi array berisi tag, atribut, dan value yang akan diproses dengan PDA.

3.4.1. def editArr(arr)

Fungsi ini menerima parameter berupa array kemudian mengubahnya sedemikian rupa sehingga dapat diproses dengan PDA yang telah kami buat. Modifikasi yang dilakukan antara lain memisahkan elemen ‘“<elemen>”’ menjadi ‘“‘, ‘<elemen>’, ‘“‘, menambahkan epsilon apabila tag img tidak memiliki atribut, menambahkan epsilon apabila tag input tidak memiliki atribut type,

mengubah semua teks kecuali tag, atribut, dan value menjadi ‘%’, serta menambahkan * ke atribut lokal.

3.4.2. def tanganiTeks(arr, item)

Fungsi ini menerima parameter berupa array dan item. Fungsi ini berfungsi untuk menambahkan value dari atribut yang valid ke dalam arr untuk membedakannya dengan value yang dapat diisi bebas.

3.4.3. handle_starttag(self, tag, attrs)

Fungsi ini menerima tiga parameter, yaitu self, tag, dan atribut. Fungsi ini berfungsi untuk mendapatkan tag-tag pembuka dari file HTML.

3.4.4. handle_endtag(self, tag)

Fungsi ini menerima dua parameter, yaitu self dan tag. Fungsi ini berfungsi untuk mendapatkan tag-tag penutup dari file HTML.

3.4.5. handle_data(self, data)

Fungsi ini menerima dua parameter, yaitu self dan data. Fungsi ini berfungsi untuk mendapatkan bagian-bagian HTML selain tag, atribut, dan value.

3.4.6. handle_comment(self, data)

Fungsi ini menerima dua parameter, yaitu self dan data. Fungsi ini berfungsi untuk mendapatkan komen-komen dalam file HTML.

BAB 5

PENGUJIAN DAN EKSPERIMEN

5.1. Testcase 1

```
<html>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a simple webpage.</p>
  </body>
  <head>
    <title>Simple Webpage</title>
  </head>
</html>
```

Gambar 5.1.1 Testcase 1

```
Syntax Error
Expected: <head> or class= or id= or style but get <body> instead
```

Gambar 5.1.2 Hasil Testcase 1

Analisis: Struktur pada file HTML tersebut tidak valid karena tag head ada di bawah tag body. Agar struktur HTML tersebut valid, maka harus diubah menjadi:

```
<html>
  <head>
    <title>Simple Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a simple webpage.</p>
  </body>
</html>
```

5.2. Testcase 2

```
<hmif>
    <head>
      <title>Simple Webpage</title>
    </head>
    <body>
      <h1>Hello, World!</h1>
      <p>This is a simple webpage.</p>
    </body>
  </hmif>
```

Gambar 5.2.1 Testcase 2

```
Syntax Error
Expected: <html> but get <head> instead
```

Gambar 5.2.2 Hasil Testcase 2

Analisis: Struktur pada file HTML tersebut tidak valid karena tidak terdapat tag <hmif> dalam HTML.. Agar struktur HTML tersebut valid, maka harus diubah menjadi:

```
<html>
  <head>
    <title>Simple Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a simple webpage.</p>
  </body>
</html>
```

5.3. Testcase 3

```
<html>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a simple webpage.</p>
  </body>
</html>
```

Gambar 5.3.1 Testcase 3

```
Syntax Error
Expected: <head> or class= or id= or style but get <body> instead
```

Gambar 5.3.2 Hasil Testcase 3

Analisis: Struktur pada file HTML tersebut tidak valid karena tidak memiliki tag head. Agar struktur HTML tersebut valid, maka seminimal mungkin harus diubah menjadi:

```
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a simple webpage.</p>
  </body>
</html>
```

5.4. Testcase 4

```
<html>
  <head>
    <title>Simple Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <h2>Welcome to my page</h2>
    <img src="./welcome.jpeg">
    <p>This is a <em>simple</em> webpage.</p>

    <div id="footer" class="footer"> This is the end of the page </div>
  </body>
</html>
```

Gambar 5.4.1 Testcase 4

Gambar 5.4.2 Hasil Testcase 4

Analisis: Struktur pada file HTML tersebut valid karena seminimalnya sudah mengandung tag html, head, dan body, beserta tag-tag penutupnya.Selain itu, penempatan value dan atribut tidak ada masalah.

5.5. Testcase 5

```html
<html>
  <head>
    <title>Simple Webpage</title>
  </head>
  <body>
    <!-- Bagian utama web -->
    <h1>Hello, World!</h1>
    <h2>Welcome to my page</h2>
    <hr>
    <img src="./welcome.jpeg" 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>
```

Gambar 5.5.1 Testcase 5

Gambar 5.5.2 Hasil Testcase 5

Analisis: Struktur pada file HTML tersebut valid karena seminimalnya sudah mengandung tag html, head, dan body, beserta tag-tag penutupnya. Selain itu, penempatan value dan atribut tidak ada masalah.

5.6.Testcase 6

```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>
```

Gambar 5.6.1 Testcase 6

```
Syntax Error
Expected: src= or alt= or class= or id= or style but get <p> instead
```

Gambar 5.6.2 Hasil Testcase 6

Analisis: Struktur pada file HTML tersebut tidak valid karena tidak ada atribut src pada tag img. Agar struktur HTML tersebut valid, maka tag img pada file tersebut harus diubah menjadi:

```
<img src=<path/link> alt="Welcome Banner">
```

5.7.Testcase 7

```html
<html>
<head>
  <title>Simple Webpage</title>

</head>
<body>

<h2>HTML Forms</h2>

<form action="/action_page.php" method="POST">
  <h5 class="label">First name:</h5><br>
  <input type="text" id="fname"><br>
  <h5 class="label">Last name:</h5><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>
```

Gambar 5.7.1 Testcase 7

Gambar 5.7.2 Hasil Testcase 7

Analisis: Struktur pada file HTML tersebut valid karena seminimalnya sudah mengandung tag html, head, dan body, beserta tag-tag penutupnya. Selain itu, penempatan value dan atribut tidak ada masalah.

5.8.Testcase 8

```
<html>
<head>
  <title>Simple Webpage</title>

</head>
<body>

<h2>HTML Forms</h2>

<form action="/action_page.php" method="TEMBAK">
  <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>
```

Gambar 5.8.1 Testcase 8

```
Syntax Error
Expected: get or post but get % instead
```

Gambar 5.8.2 Hasil Testcase 8

Analisis: Struktur pada file HTML tersebut tidak valid karena value pada atribut form tidak valid. Agar valid, kata “TEMBAK” pada method form harus diganti dengan POST atau GET.

5.9.Testcase 9

```html
<html>
<head>
  <title>Simple Webpage</title>
  <script>
    document.getElementById("demo").innerHTML = "Hello JavaScript!";
  </script>
</head>
<body>

<h1>The script element</h1>

<p id="demo"></p>

</body>
</html>
```

Gambar 5.9.1 Testcase 9

Gambar 5.9.2 Hasil Testcase 9

Analisis: Struktur pada file HTML tersebut valid karena seminimalnya sudah mengandung tag html, head, dan body, beserta tag-tag penutupnya. Selain itu, penempatan value dan atribut tidak ada masalah.

5.10. Testcase 10

```
<html>
<head>
  <title>Simple Webpage</title>
  <script>
    document.getElementById("demo").innerHTML = "Hello JavaScript!";
  </script>
</head>
<body>

<h1>The script element</h1>

<p id="demo">

</body>
</html>
```

Gambar 5.10.1 Testcase 10

```
Syntax Error
Expected: % or </p> or <em> or <b> or <abbr> or <strong> or <small> or class= or id= or style but get </body> instead
```

Gambar 5.10.2 Hasil Testcase 10

Analisis: Struktur pada file HTML tersebut tidak valid karena tidak ada tag penutup p. Agar menjadi valid, tag p seharusnya diubah menjadi

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

5.11. Testcase 11

```html
<html>
    <head>
        <title>Simple Webpage</title>
        <script>
            document.getElementById("demo").innerHTML = "Hello JavaScript!";
        </script>
    </head>
    <body>
        <h1>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>
```

Gambar 5.11.1 Testcase 11

Gambar 5.11.2 Hasil Testcase 11

Analisis: Struktur pada file HTML tersebut valid karena seminimalnya sudah mengandung tag html, head, dan body, beserta tag-tag penutupnya. Selain itu, penempatan value dan atribut tidak ada masalah.

5.12. Testcase 12

```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Random HTML Page</title>

</head>
<body>

    <header>
        <h1>Welcome to My Random Page!</h1>
    </header>

    <section>
        <p>This is a random paragraph of text. It doesn't really mean anything, but it's here to fill space on the page.</p>

        <h2>Random List:</h2>

    </section>

    <section>
        <p>Here's another random section. Feel free to replace this with your own content!</p>
    </section>

</body>
</html>
```

Gambar 5.12.1 Testcase 12

Gambar 5.12.2 Hasil Testcase 12

Analisis: Struktur pada file HTML tersebut valid karena seminimalnya sudah mengandung tag html, head, dan body, beserta tag-tag penutupnya. Selain itu, penempatan value dan atribut tidak ada masalah.

5.13. Testcase 13

```html
<html>
<head>
    <title>Complex HTML Page</title>
</head>
<body>
    <header>
        <h1>Welcome to My Complex Page!</h1>
    </header>
    <div>
        <p>This is a more complex example with a navigation bar, form, and some interactivity.</p>
        <h2>Random List</h2>
    </div>
        <form>
            <input type="text" id="name">
            <br>
            <input type="email" id="email">
            <br>
            <input type="submit">
        </form>
    <script>
        // Just a simple example of JavaScript for interactivity
        document.querySelector('h1').addEventListener('click', function() {
            alert('You clicked the heading!');
        });
    </script>
</body>
</html>
```

Gambar 5.13.1 Testcase 13

```
Syntax Error
Expected: text or password or email or number or checkbox but get submit instead
```

Gambar 5.13.2 Hasil Testcase 13

Analisis: Struktur pada file HTML tersebut tidak valid karena atribut type dari tag input tidak menerima value submit. Atribut tersebut hanya menerima value text, password, email, number, dan checkbox.

BAB 6

KESIMPULAN

6.1. Kesimpulan

Dari tugas besar TBFO kali ini, dapat disimpulkan bahwa pushdown automata (PDA) dapat digunakan untuk mengecek kevalidan struktur HTML. Namun, masih terdapat beberapa keterbatasan-keterbatasan, antara lain formatting string tidak dapat dilakukan di dalam tag **form**, **button**, dan **a.**
Keterbatasan-keterbatasan lain dapat dilihat pada asumsi dalam bab 3.

6.2. Saran

Untuk pembuatan yang lebih mudah dengan state yang lebih sedikit, dapat digunakan arbitrary variabel.

BAB 7

Lain-Lain

7.1. Pembagian Tugas

NIM	Nama	Pembagian Tugas
13522083	Evelyn Yosiana	HTML parser, laporan, readme.
13522090	Fedrianz Dharma	Main program, parser, laporan.
13522103	Steven Tjhia	PDA.

LAMPIRAN

Link github:

https://github.com/FedrianzD/Tugas-Besar-TBFO.git

Link desain PDA:

https://www.figma.com/file/xnO6KGqNHYtAqvwSg7hHrA/Tugas-Besar-FO?type=whiteboard&node-id=0%3A1&t=uFIpL5Kvjgj4aJPB-1

DAFTAR PUSTAKA

Hopcroft, John. Automata Theory, Language, and Computation 3rd Edition. London: Pearson Education

W3Schools. HTML Tag. Diakses pada 22 November 2023 dari https://www.w3schools.com/html