Tugas Pemrograman
IF2124 Teori Bahasa Formal dan Otomata
HTML Checker dengan Pushdown Automata (PDA)
Semester I Tahun 2023/2024

Disusun Oleh:

Ariel Herfrison	(13522002)
Irfan Sidiq Permana	(13522007)
Bryan Cornelius Lauwrence	(13522033)

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG

2023

BAB 1
TEORI DASAR

Pushdown Automata adalah otomata yang mendefinisikan Context-Free Language. Pada intinya, Pushdown Automata adalah ϵ-NFA yang diperlengkapi dengan stack. Pushdown Automata dapat menerima input berdasarkan *accepting state* atau berdasarkan stack yang kosong. Salah satu subset PDA adalah deterministic PDA. Sebuah PDA adalah deterministic PDA jika dan hanya jika setiap δ(q, a, X) adalah kosong atau *singleton* dan jika δ(q, a, X) tidak kosong, maka δ(q, ϵ, X) harus kosong. PDA dapat menerima kondisi akhir berdasarkan *stack* kosong atau *state* yang ditentukan sebagai *state* akhir (*final state*).

PDA dapat dimanfaatkan sebagai *bug finder*, salah satunya adalah *bug finder* untuk HTML. HTML (Hypertext Markup Language) adalah salah satu *markup language* untuk menyusun sebuah website dan kini digunakan oleh hampir semua website yang ada di internet. Sintaks HTML terdiri dari elemen tag yang dapat berupa elemen yang memiliki tag tutup, atau elemen yang tidak memiliki tag tutup (*void element*). Tiap elemen tag dapat memiliki suatu atribut global yang dapat dimiliki semua tag, misalnya *id*, *class*, *style*, dan sebagainya, serta atribut khusus yang hanya dimiliki oleh tag tertentu, misalnya *src*, *href*, dan sebagainya.

Untuk mengecek error atau tidaknya suatu file HTML, PDA sebagai HTML *bug finder* yang dibuat akan membaca elemen tag, lalu menyimpannya di dalam *stack* dan mengeluarkannya dari *stack* jika ditemukan tag penutupnya. Khusus *void element*, PDA akan mengeluarkan tag dari stack setelah menerima karakter '>' dari tag pembuka. PDA juga akan berpindah-pindah *state* bergantung posisi pengecekan saat ini. Pengecekan atribut suatu tag dilakukan dengan membaca tag buka terlebih dahulu lalu membaca atributnya jika ada. Kemudian, PDA akan mengecek atribut tersebut valid atau tidaknya atribut tersebut.

BAB 2
HASIL PDA

Berikut adalah format definisi PDA (PDA menerima *final state*)

```
Q P F # total states
a # input word symbols
Z Y # stack symbols
Q # starting state
Z # starting stack
F # accepting states
F # E - accepts with empty stack or F - accepts with accepting
state
Q a Z Q Y,Z # list of productions (current state, read from
word, take from stack, next state, add to stack)
Q a Y Q Y,Y
Q e Z P Z
Q e Y P Y
P a Z P e
P a Y P e
P e Z F e
```

Berikut adalah hasil definisi PDA HTML Checker:

```
Q0 H B F Text UT UAttr Dead
" < > / id=" class=" style=" <html </html> <head </head> <title
title> <script src=" script> <link href=" rel= <body </body>
<h1 h1 h1> <h2 h2 h2> <h3 h3 h3> <h4 h4 h4> <h5 h5 h5> <h6 h6
h6> <a a a> <img img alt=" src= <script script> <link <p p p>
<br br <hr hr <em em em> <b b r utton b> <abbr abbr abbr>
<strong strong strong> <small small small> <div div div> </div>
<button button type="submit" type="reset" type="button" button>
<form form form> <input input type="text" type="password"
type="email" type="number" type="checkbox" <table table
action=" method="POST" method="GET" </table> <tr </tr> <th th>
<td td> <!- !- -
Z0 Ga GaT Html Head Body Title HScript BScript HLinkB HLinkA
BLinkB BLinkA H1 H2 H3 H4 H5 H6 P Br Em Bold Abbr Strong Small
Hr Div A ImgB ImgA  Button Form    FormN Input Table TR TD TH K
KQ0 KH KB KF KUT KA
Q0
Z0
F
F
```

```
Q0    <!-   %     Text KQ0,%
H     <!-   %     Text KH,%
B     <!-   %     Text KB,%
F    <    %    Dead     %
Dead !-    %     Text KF,%
UT    !-    %     Text KUT,%
Text -     K     Text KA
Text %     K     Text K
Text -     KA    Text e
Text %     KA    Text K
Text >     KQ0   Q0    e
Text >     KH    H     e
Text >     KB    B     e
Text >     KF    F     e
Text >     KUT   Text e
Text %     KQ0   Text K,KQ0
Text %     KH    Text K,KH
Text %     KB    Text K,KB
Text %     KF    Text K,KF
Text %     KUT   Text K,KUT
Text "     Ga    UAttr     e
Text %     Ga    Text Ga
Text <     GaT       UT         GaT
Text <     FormN      UT        FormN
Text     <    DivN     UT       DivN
Text %     GaT       Text  GaT
Text %     FormN      Text FormN
Text     %    DivN     Text     DivN
UAttr      id=" %     Text Ga,%
UAttr      class="    %     Text Ga,%
UAttr      style="    %     Text Ga,%
UAttr      >     Html Q0        Html
UAttr      >     Head H         Head
UAttr      >     Body B         Body
UT        /      GaT       UT         e
Q0        <html Z0    UAttr      Html,Z0
Q0    <head      Html UAttr      Head,Html
H     <title     Head UAttr      Title,Head
UAttr      >     Title      Text GaT,Title
UAttr      >     HScript    Text GaT,HScript
UT    title>     Title      H     e
H     <script    Head UAttr      HScript,Head
UAttr      src="      HScript    Text Ga,HScript
UT    script>    HScript    H     e
H     <link      Head UAttr      HLinkB,Head
UAttr      href="     HLinkB     Text Ga,HLinkB
```

```
UAttr     rel= HLinkB    UAttr     HLinkA
UAttr     "    HLinkA    Text Ga,HLinkA
UAttr     href="    HLinkA    Text Ga,HLinkA
UAttr     >    HLinkA    H    e
H    </head>   Head H    e
H    <body     Html UAttr     Body,Html
B    <h1  Body UAttr     H1,Body
UT   h1   DivN UAttr     H1,DivN
UT   h1   FormN     UAttr     H1,FormN
UAttr     >    H1   Text GaT,H1
UT   h1>  H1   B    e
B    <h2  Body UAttr     H2,Body
UT   h2   DivN UAttr     H2,DivN
UT   h2   FormN     UAttr     H2,FormN
UAttr     >    H2   Text GaT,H2
UT   h2>  H2   B    e
B    <h3  Body UAttr     H3,Body
UT   h3   DivN UAttr     H3,DivN
UT   h3   FormN     UAttr     H3,FormN
UAttr     >    H3   Text GaT,H3
UT   h3>  H3   B    e
B    <h4  Body UAttr     H4,Body
UT   h4   DivN UAttr     H4,DivN
UT   h4   FormN     UAttr     H4,FormN
UAttr     >    H4   Text GaT,H4
UT   h4>  H4   B    e
B    <h5  Body UAttr     H5,Body
UT   h5   DivN UAttr     H5,DivN
UT   h5   FormN     UAttr     H5,FormN
UAttr     >    H5   Text GaT,H5
UT   h5>  H5   B    e
B    <h6  Body UAttr     H6,Body
UT   h6   DivN UAttr     H6,DivN
UT   h6   FormN     UAttr     H6,FormN
UAttr     >    H6   Text GaT,H6
UT   h6>  H6   B    e
B    <a   Body UAttr     A,Body
UT   a    DivN UAttr     A,DivN
UT   a    FormN     UAttr     A,FormN
UAttr     href="    A    Text Ga,A
UAttr     >    A    Text GaT,A
UT   a>   A    B    e
B    <img Body UAttr     ImgB,Body
UT   img  DivN UAttr     ImgB,DivN
UT   img  FormN     UAttr     ImgB,FormN
UAttr     alt="     ImgB Text Ga,ImgB
```

```
UAttr     src= ImgB UAttr     ImgA
UAttr     "    ImgA Text Ga,ImgA
UAttr     alt="     ImgA Text Ga,ImgA
UAttr     >    ImgA B    e
B    <script   Body UAttr     BScript,Body
UT   script    DivN UAttr     Bscript,DivN
UT   script    FormN     UAttr     BScript,FormN
UAttr     src="     BScript   Text Ga,BScript
UT   script>   BScript   B    e
B    <link     Body UAttr     BLinkB,Body
UT   link DivN UAttr     BLinkB,DivN
UT   link FormN     UAttr     BLinkB,FormN
UAttr     href="    BLinkB    Text Ga,BLinkB
UAttr     rel= BLinkB    UAttr     BLinkA
UAttr     "    BLinkA    Text Ga,BLinkA
UAttr     href="    BLinkA    Text Ga,BLinkA
UAttr     >    BLinkA    B    e
B    <p   Body UAttr     P,Body
UT   p    DivN UAttr     P,DivN
UT   p    FormN     UAttr     P,FormN
UAttr     >    P    Text GaT,P
UT   p>   P    B    e
B    <br  Body UAttr     Br,Body
UAttr     >    Br   B    e
B   e   GaT Text    GaT
B    <hr  Body UAttr     Hr,Body
UT   hr   DivN UAttr     Hr,DivN
UT   hr   FormN     UAttr     Hr,FormN
UAttr     >    Hr   B    e
UT   em   GaT  UAttr     Em,GaT
UT  em  DivN    UAttr   Em,DivN
UT   em   FormN     UAttr     Em,FormN
UAttr     >    Em   Text GaT,Em
UT   em>  Em   Text e
UT   b    GaT  UAttr     Bold,GaT
UAttr  r   Bold    UAttr    Br
UAttr utton Bold    UAttr   Button
UT   b    FormN     UAttr     Bold,FormN
UT  b   DivN    UAttr   Bold,DivN
UAttr     >    Bold Text GaT,Bold
UT   b>   Bold Text e
UT   abbr GaT      UAttr Abbr,GaT
UT   abbr FormN     UAttr     Abbr,FormN
UT  abbr    DivN    UAttr   Abbr,DivN
UAttr     >    Abbr Text GaT,Abbr
UT   abbr>     Abbr Text e
```

```
UT   strong    GaT      UAttr Strong,GaT
UT   strong    FormN     UAttr     Strong,FormN
UT  strong  DivN    UAttr   Strong,DivN
UAttr     >    Strong    Text GaT,Strong
UT   strong>   Strong    Text e
UT   small     GaT      UAttr Small,GaT
UT   small     FormN     UAttr     Small,FormN
UT  small   DivN    UAttr   Small,DivN
UAttr     >    Small     Text GaT,Small
UT   small>    Small     Text e
B    <div Body UAttr     Div,Body
UT   div  DivN     UAttr Div,DivN
UT   div       FormN UAttr     Div,FormN
UT   div       DivN  UAttr     Div,DivN
UAttr     >    Div       Text    DivN,Div
UT  /       DivN    UT      e
UT   div> Div      B          e
B   e       DivN    Text    DivN
B    <button   Body UAttr     Button,Body
UAttr     >    Button    Text GaT,Button
UAttr     type="submit"  Button    UAttr     Button
UAttr     type="reset"   Button    UAttr     Button
UAttr     type="button"  Button    UAttr     Button
UT   button>   Button    B    e
B    <form     Body UAttr     Form,Body
UT  form   DivN     UAttr   Form,DivN
UT   div  FormN     UAttr     Div,FormN
UAttr     method="GET"   Form UAttr     Form
UAttr     method="POST"  Form UAttr     Form
UAttr     action="  Form Text Ga,Form
UAttr     >    Form Text FormN,Form
UT   /    FormN     UT   e
UT   form>     Form B    e
B    e    FormN     Text FormN
B    <input    Body UAttr     Input,Body
UT   input     DivN UAttr     Input,DivN
UT   input     FormN     UAttr     Input,FormN
UAttr     type="text"    Input     UAttr     Input
UAttr     type="password"     Input     UAttr     Input
UAttr     type="email"   Input     UAttr     Input
UAttr     type="number"  Input     UAttr     Input
UAttr     type="checkbox"     Input     UAttr     Input
UAttr     >    Input     B    e
B    <table    Body UAttr     Table,Body
UT   table     FormN     UAttr     Table,FormN
UT  table   DivN    UAttr   Table,DivN
```

```
UAttr      >     Table      B     Table
B     <tr  Table      UAttr      TR,Table
UAttr      >     TR    B     TR
B     <th  TR    UAttr      TH,TR
B     <td  TR    UAttr      TD,TR
UAttr      >     TH    Text  GaT,TH
UAttr      >     TD    Text  GaT,TD
UT    th>  TH    B     e
UT    td>  TD    B     e
B     </tr>      TR    B     e
B     </table>   Table      B     e
B     </body>    Body  B     e
B     </html>    Html  F     e
```

Berikut adalah program PDA dalam python

```python
import os
import argparse

state_list = []
input_list = []
stack_symbols = []
stack = []
global state
final_states = []
accept_condition = None
transition_table = []

def printPda():
    print(state_list)
    print(input_list)
    print(stack_symbols)
    print(stack)
    print(state)
    print(final_states)
    print(accept_condition)
    for i in transition_table:
        print(i)

def inputAccepted(input, state, stack):
    for i in transition_table:
        if (i[0] == state and i[1] == input and stack[-1] == i[2]):
            return True
    return False

def splitSymbols(string):
    temp =  string.split(',')
    temp.reverse()
```

```
    return temp

def epsilonTransition(state,stack):
    can_epsilon = True
    while can_epsilon:
        can_epsilon = False
        if stack != []:
            for i in transition_table:
                if i[0] == state and i[1] == 'e' and stack[-1] == i[2]:
                    state = i[3]
                    stack.pop()
                    for j in splitSymbols(i[4]):
                        stack.append(j)
                    can_epsilon = True
                    break
    return (state,stack)

dirname = os.path.dirname(__file__)

parser = argparse.ArgumentParser()
parser.add_argument("a")
parser.add_argument("b")
args = parser.parse_args()

nama_file_pda = args.a
nama_file_pda = os.path.join(dirname,nama_file_pda)
file_pda = open(nama_file_pda,"r")
#print(file_pda)

nama_file_html = args.b
nama_file_html = os.path.join(dirname,nama_file_html)
file_html = open(nama_file_html,"r")

# convert pda from txt to py
for word in file_pda.readline().split():
    state_list.append(word)
for word in file_pda.readline().split():
    input_list.append(word)
for word in file_pda.readline().split():
    stack_symbols.append(word)
state = file_pda.readline().split()[0]
for word in file_pda.readline().split():
    stack.append(word)
for word in file_pda.readline().split():
    final_states.append(word)
accept_condition = file_pda.readline().split()[0]
for line in file_pda:
    if (line.strip()):
        transition_table.append(line.split())
#printPda()
```

```
Lines = file_html.readlines()

word_buffer = ""
m=1
last_input = m
last_input_char = 0
for line in Lines:
    n=0
    for char in line:
        n += 1
        input = None
        state, stack = epsilonTransition(state,stack)
        old_state = state[:]
        old_stack = stack[:]

        if char != ' ' and char != '\n':
            word_buffer += char
            # print(word_buffer)

        if word_buffer in input_list:
            input = word_buffer

        if input != None:
            can_search = False
            if stack != []:
                for i in transition_table:
                    if i[0] == state and i[1] == input and (stack[-1] == i[2] or i[2] ==
'%'):
                        state = i[3]
                        temp = stack[-1]
                        stack.pop()
                        if i[4] != 'e':
                            for j in splitSymbols(i[4]):
                                if j == '%':
                                    stack.append(temp)
                                else:
                                    stack.append(j)
                        can_search = True
                        break
            if can_search:
                last_input = m
                last_input_char = n
                word_buffer = ''
                # print(input,old_state,old_stack,state,stack)
                # print(input,m,state,stack)
            # else:
            #     print(input)

        if input == None or not can_search:
            # '%' sebagai pengganti simbol 'all'
            for i in transition_table:
```

```
                if i[0] == state and i[1] == '%' and (stack[-1] == i[2] or i[2] == '%'):
                    word_buffer = ""
                    state = i[3]
                    temp = stack[-1]
                    stack.pop()
                    if i[4] != 'e':
                        for j in splitSymbols(i[4]):
                            # print(j)
                            if j == '%':
                                stack.append(temp)
                            else:
                                stack.append(j)
                    break
    m += 1

epsilonTransition(state,stack)
# print(state,stack)
if (accept_condition == "F" and state in final_states) or (accept_condition == "E" and stack
== []):
    print("Accepted")
else:
    print("Syntax Error")
    temp_line = Lines[last_input-1]
    print(f"Terjadi kesalahan ekspresi pada line {last_input} :
'{temp_line[:last_input_char].rstrip()+"\033[4m"+temp_line[last_input_char:].rstrip()+"\033[
0m"}' karakter ke-{last_input_char+1} : '{temp_line[last_input_char].rstrip()}'")
    temp = []
    for i in transition_table:
        if i[0] == state and i[2] == stack[-1] and splitSymbols(i[4])[0] != stack[-1]:
            temp.append(i[1])
    if len(temp)>0:
        print("Expected input: ",end="")
        for i in range(len(temp)):
            if (i>0):
                print(", ",end="")
            print(temp[i],end=" ")
    else:
        temp = []
        for i in transition_table:
            if i[0] == state and i[2] == stack[-1]:
                temp.append(i[1])
        if len(temp)>0:
            print("Expected input: ",end="")
            for i in range(len(temp)):
                if (i>0):
                    print("|| ",end="")
                print(temp[i],end=" ")
```

BAB 3
IMPLEMENTASI DAN PENGUJIAN

1. Spesifikasi Teknis Program

Program dijalankan dalam command line dengan memanggil program beserta 2 argumen. Argumen pertama berupa text file (.txt) berisi definisi PDA. Argumen kedua berupa html file (.html) berisi kode HTML. Program kemudian akan membaca text file dan mencatat list states, list input word symbols, list stack symbols, starting state, starting stack, accepting states, accept condition, dan list of productions. Kemudian, program akan membaca html file dan menjalankan algoritma PDA untuk mencari tahu apakah html diterima atau tidak. Jika tidak diterima, program akan menunjukkan line di mana kode tidak sesuai dan menunjukkan expected input yang seharusnya diberikan. Pada program html checker kami, program akan melakukan epsilon transition terlebih dahulu sebelum lanjut memroses input (karakter pada file html).

2. Hasil Pengujian

1. Test Case 1

Expected Output: **Rejected** (head muncul setelah body)

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

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Syntax Error
Terjadi kesalahan ekspresi pada line 1 : '<html>' karakter ke-7 : ''
Expected input: <head
```

2. Expected Output: **Rejected** (tag pertama bukan html)

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

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Syntax Error
Terjadi kesalahan ekspresi pada line 1 : '<hmif>' karakter ke-1 : '<'
Expected input: <html
```

3. Expected Output: **Rejected** (tidak punya head)

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

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Syntax Error
Terjadi kesalahan ekspresi pada line 1 : '<html>' karakter ke-7 : ''
Expected input: <head
```

4. **Accepted**

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Accepted
```

5. Expected Output: **Accepted**

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Accepted
```

6. Expected Output: **Rejected (**Atribut wajib src pada img tidak ada)

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Syntax Error
Terjadi kesalahan ekspresi pada line 9 : '    <img alt="Welcome Banner">' karakter ke-30 : '>'
Expected input: src=
```

7. Expected Output: **Accepted**

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Accepted
```

8. Expected Output: **Rejected (**tidak termasuk value yang diperbolehkan (POST, GET))

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Syntax Error
Terjadi kesalahan ekspresi pada line 10 : '<form action="/action_page.php" method="TEMBAK">' karakter ke-32 : ''
Expected input: method="GET" || method="POST" || action=" || >
```

9. Expected Output: **Accepted**

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Accepted
```

10. Expected Output: **Rejected** (<p> bukan void element, wajib ditutup)

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Syntax Error
Terjadi kesalahan ekspresi pada line 14 : '</body>' karakter ke-3 : 'b'
Expected input: p>
```

11. Expected Output: **Accepted**

```
<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>
```

Output:

```
PS C:\Ariel\Kuliah\MatKul\Tingkat 2\TBFO\Tubes\Tubes-IF2124-TBFO-2023-2024\src> python pda_machine.py PDA.txt tes_html.html
Accepted
```

REFERENSI

Link repository github: github.com/BryanLauw/Tubes-IF2124-TBFO-2023-2024

Link diagram state: https://tinyurl.com/37eer7u4

Pembagian Tugas:

NIM	NAMA	PEMBAGIAN TUGAS
13522002	Ariel Herfrison	Membuat fungsi PDA dalam python
13522007	Irfan Sidiq P	Merumuskan definisi PDA
13522033	Bryan Cornelius	Merumuskan definisi PDA