Tugas Besar IF2124 Teori Bahasa Formal dan Otomata

HTML *Checker* dengan Pushdown Automata

Semester I Tahun 2023/2024

Disusun oleh:

Devinzen (13522064)

Matthew Vladimir Hutabarat (13522093)

Berto Richardo Togatorop (13522118)

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2023

Daftar Isi

BAB I

Deskripsi Masalah

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

Bab II

Landasan Teori

2.1 HTML

HTML merupakan singkatan dari HyperText Markup Language. Bahasa ini sering digunakan dalam web development untuk membuat kerangka website. HTML memberikan struktur dasar pada website dengan menggunakan elemen-elemen markup atau tag untuk mengidentifikasi bagian-bagian berbeda dari halaman web. Setiap tag HTML memiliki fungsi tertentu dan mendefinisikan elemen-elemen dan struktur konten. Biasanya HTML digunakan bersama dengan CSS. Poin-poin utama pada HTML ialah:

1. **Elemen HTML:** Elemen-elemen HTML adalah bagian dasar dari website seperti judul, paragraf, gambar, tautan, formulir, tabel, dan lain-lain . Setiap elemen memiliki sintaks dan atribut sendiri.
2. **Tag HTML:** Tag HTML digunakan untuk menunjukan fungsi atau arti tertentu. Di HTML ada banyak sekali Tag, contohnya <p> digunakan untuk paragraf>, <img> digunakan untuk menampilkan gambar, <a> digunakan untuk menentukan url tautan, dan lain-lain.
3. **Atribut HTML:** Atribut berada pada tag HTML, Atribut digunakan untuk memberikan informasi tambahan seperti pada tag <img> terdapat atribut ‘src’ yang berfungsi untuk memberikan sumber gambar diambil.
4. **Struktur Dasar HTML:** Struktur dasar halaman HTML terdiri dari elemen seperti <html> , <head>, dan <body> . Elemen <html> menyatakan awal dan akhir dokumen HTML, <head> berisi informasi seperti judul dan tautan ke CSS, sedangkan <body> berisi konten yang ditampilkan di halaman website.

2.2 Push Down Automata (PDA)

PDA atau Pushdown Automaton adalah salah satu pelajaran yang diajarkan di kelas Tata Bahasa Formal Automata. PDA banyak digunakan untuk memodelkan kelompok bahasa kontekstual, yang merupakan kelas bahasa dalam teori bahasa formal yang lebih kuat daripada bahasa reguler. Beberapa penerapan PDA ialah:

1. Pengenalan dan Penerapan Bahasa Kontekstual
2. Kompilasi dan Analisis Sintaksis
3. Validasi dan Parsing Dokumen
4. Pengenalan Pola dan String
5. Modelling Sistem Formal
6. Pengembangan Algoritma
7. Analisis Struktur Data

Adapun deskripsi PDA yang kebih rinci seperti:

1. **Komponen Utama:**
    a. Alfabet Input : Sekumpulan simbol yang dapat dibaca oleh PDA
    b. Alfabet Tumpukan : Sekumpulan simbol yang dapat ditempatkan di tumpukan
    c. Kondisi (State): Sekumpulan kondisi atau status yang dapat ditemukan PDA
    d. Fungsi Transisi : Aturan yang menentukan bagaimana PDA bergerak dari satu kondisi ke kondisi lainnya bedasarkan simbol input dan simbol tumpukan saat ini.
2. **Tumpukan (Stack):** PDA memiliki tumpukan (stack) sebagai penyimpanan tambahan untuk membantu dalam pengenalan bahasa kontekstual
3. **Fungsi Transisi :** Fungsi transisi menggambarkan bagaimana PDA berpindah dari satu kondisi ke kondisi lainnya bedasarkan simbol input saat ini, simbol tumpukan saat ini, dan simbol yang ditumpuk atau dihapus dari tumpukan. Fungsi transisi ini mendefinisikan aturan-aturan formal yang mengatur perilaku PDAQ

Bab III

Hasil PDA

3.1. States

Z, Q, H, HC, B, BC, BOQ, F, TC, ATT, ATTCONTENT, ATTFORM , ATTBUTTON ATTINPUT, C

3.2. Input Aphabet

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

3.3. Stack Alphabet

Html, HeadHtml, BodyHeadHtml, Class, Id, Style, Head, Html, Class, Id, Style, Head, Body, Title, ScriptHead, ScriptBody, LinkHead, LinkBody, RelationLinkHead, RelationLinkBody, Relation, HypertextReference, Source, H1, H2, H3, H4, H5, H6, P, Br, Em, B, Abbr, Strong, Small, Hr, Div, A, Img, SourceImg, Alternative, Type, TypeInput, Input, Button, Form, Table, TableData, TableHeader, TableRow

3.4. Transition Function

δ(ATT, class=", $) = {(ATTCONTENT, Class$)}
δ(ATT, id=", $) = {(ATTCONTENT, Id$)}
δ(ATT, style=", $) = {(ATTCONTENT, Style$)}

δ(ATTCONTENT, %", $) = {(ATT, e)}
δ(C, %, $) = {(C, $)}
δ(BC, %, Form) = {(BC, Form)}
δ(BC, %, MethodForm) = {(BC, MethodForm)}
δ(BC, %, Div) = {(BC, Div)}
δ(Q, &%, Z) = {(Q, Z)}
δ(C, &%, $) = {(C, $)}
δ(HC, &%, $) = {(HC, $)}
δ(BC, &%, $) = {(BC, $)}
δ(BOQ, &%, $) = {(BOQ, $)}
δ(TC, &%, $) = {(TC, $)}
δ(F, &%, Z) = {(F, Z)}
δ(Q, <html, Z) = {(ATT, HtmlZ)}
δ(ATT, >, Html) = {(H, Html)}
δ(BOQ, </html>, BodyHeadHtml) = {(F, e)}
δ(H, <head, Html) = {(ATT, HeadHtml)}
δ(ATT, >, HeadHtml) = {(HC, HeadHeadHtml)}
δ(HC, </head>, Head) = {(B, e)}
δ(HC, <title, Head) = {(ATT, TitleHead)}
δ(ATT, >, Title) = {(C, Title)}
δ(C, </title>, Title) = {(HC, e)}
δ(HC, <script, Head) = {(ATT, ScriptHeadHead)}
δ(BC, <script, Body) = {(ATT, ScriptBodyBody)}
δ(ATT, src=", ScriptHead) = {(ATTCONTENT,
SourceScriptHead)}
δ(ATT, src=", ScriptBody) = {(ATTCONTENT,
SourceScriptBody)}
δ(ATT, >, ScriptHead) = {(C, ScriptHead)}
δ(ATT, >, ScriptBody) = {(BC, ScriptBody)}
δ(C, </script>, ScriptHead) = {(HC, e)}
δ(C, </script, ScriptBody) = {(BC, e)}
δ(HC, <link, Head) = {(ATT, LinkHeadHead)}
δ(BC, <link, Body) = {(ATT, LinkBodyBody)}
δ(BC, <link, Div) = {(ATT, LinkBodyDiv)}
δ(ATT, rel=", LinkHead) = {(ATTCONTENT,
RelationRelationLinkHead)}
δ(ATT, rel=", LinkBody) = {(ATTCONTENT,
RelationRelationLinkBody)}
δ(ATT, href=", LinkHead) = {(ATTCONTENT,
HypertextReferenceLinkHead)}

δ(ATT, href=", LinkBody) = {(ATTCONTENT, HypertextReferenceLinkBody)}
δ(ATT, href=", RelationLinkHead) = {(ATTCONTENT, HypertextReferenceRelationLinkHead)}
δ(ATT, href=", RelationLinkBody) = {(ATTCONTENT, HypertextReferenceRelationLinkBody)}
δ(ATT, >, RelationLinkHead) = {(HC, e)}
δ(ATT, >, RelationLinkBody) = {(BC, e)}
δ(B, <body, HeadHtml) = {(ATT, BodyHeadHtml)}
δ(ATT, >, BodyHeadHtml) = {(BC, BodyBodyHeadHtml)}
δ(BC, </body>, Body) = {(BOQ, e)}
δ(BC, <h1, Body) = {(ATT, H1Body)}
δ(BC, <h2, Body) = {(ATT, H2Body)}
δ(BC, <h3, Body) = {(ATT, H3Body)}
δ(BC, <h4, Body) = {(ATT, H4Body)}
δ(BC, <h5, Body) = {(ATT, H5Body)}
δ(BC, <h6, Body) = {(ATT, H6Body)}
δ(BC, <h1, Div) = {(ATT, H1Div)}
δ(BC, <h2, Div) = {(ATT, H2Div)}
δ(BC, <h3, Div) = {(ATT, H3Div)}
δ(BC, <h4, Div) = {(ATT, H4Div)}
δ(BC, <h5, Div) = {(ATT, H5Div)}
δ(BC, <h6, Div) = {(ATT, H6Div)}
δ(BC, <h1, Form) = {(ATT, H1Form)}
δ(BC, <h2, Form) = {(ATT, H2Form)}
δ(BC, <h3, Form) = {(ATT, H3Form)}
δ(BC, <h4, Form) = {(ATT, H4Form)}
δ(BC, <h5, Form) = {(ATT, H5Form)}
δ(BC, <h6, Form) = {(ATT, H6Form)}
δ(BC, <h1, MethodForm) = {(ATT, H1MethodForm)}
δ(BC, <h2, MethodForm) = {(ATT, H2MethodForm)}
δ(BC, <h3, MethodForm) = {(ATT, H3MethodForm)}
δ(BC, <h4, MethodForm) = {(ATT, H4MethodForm)}
δ(BC, <h5, MethodForm) = {(ATT, H5MethodForm)}
δ(BC, <h6, MethodForm) = {(ATT, H6MethodForm)}
δ(ATT, >, H1) = {(C, H1)}
δ(ATT, >, H2) = {(C, H2)}
δ(ATT, >, H3) = {(C, H3)}
δ(ATT, >, H4) = {(C, H4)}
δ(ATT, >, H5) = {(C, H5)}
δ(ATT, >, H6) = {(C, H6)}

δ(C, </h1>, H1) = {(BC, e)}
δ(C, </h2>, H2) = {(BC, e)}
δ(C, </h3>, H3) = {(BC, e)}
δ(C, </h4>, H4) = {(BC, e)}
δ(C, </h5>, H5) = {(BC, e)}
δ(C, </h6>, H6) = {(BC, e)}
δ(BC, <p, Body) = {(ATT, PBody)}
δ(BC, <p, Div) = {(ATT, PDiv)}
δ(BC, <p, Form) = {(ATT, PForm)}
δ(BC, <p, MethodForm) = {(ATT, PMethodForm)}
δ(ATT, >, P) = {(C, P)}
δ(C, </p>, P) = {(BC, e)}
δ(BC, <br, Body) = {(ATT, BrBodyBody)}
δ(C, <br, P) = {(ATT, Br)}
δ(C, <br, H1) = {(ATT, BrH1)}
δ(C, <br, H2) = {(ATT, BrH2)}
δ(C, <br, H3) = {(ATT, BrH3)}
δ(C, <br, H4) = {(ATT, BrH4)}
δ(C, <br, H5) = {(ATT, BrH5)}
δ(C, <br, H6) = {(ATT, BrH6)}
δ(C, <br, A) = {(ATT, BrA)}
δ(C, <br, button) = {(ATT, BrButton)}
δ(BC, <br, Div) = {(ATT, BrDivDiv)}
δ(BC, <br, Form) = {(ATT, BrFormForm)}
δ(BC, <br, MethodForm) = {(ATT, BrMethodFormMethodForm)}
δ(ATT, >, BrBody) = {(BC, e)}
δ(ATT, >, BrDiv) = {(BC, e)}
δ(ATT, >, BrForm) = {(BC, e)}
δ(ATT, >, BrMethodForm) = {(BC, e)}
δ(ATT, >, Br) = {(C, e)}
δ(BC, <em, Body) = {(ATT, EmBodyBody)}
δ(C, <em, P) = {(ATT, Em)}
δ(C, <em, H1) = {(ATT, EmH1)}
δ(C, <em, H2) = {(ATT, EmH2)}
δ(C, <em, H3) = {(ATT, EmH3)}
δ(C, <em, H4) = {(ATT, EmH4)}
δ(C, <em, H5) = {(ATT, EmH5)}
δ(C, <em, H6) = {(ATT, EmH6)}
δ(C, <em, A) = {(ATT, EmA)}
δ(C, <em, button) = {(ATT, EmButton)}
δ(BC, <em, Div) = {(ATT, EmDivDiv)}

δ(BC, <em, Form) = {(ATT, EmFormForm)}
δ(BC, <em, MethodForm) = {(ATT, EmMethodFormMethodForm)}
δ(ATT, >, EmBody) = {(BC, EmBody)}
δ(ATT, >, EmDiv) = {(BC, EmDiv)}
δ(ATT, >, EmForm) = {(BC, EmForm)}
δ(ATT, >, EmMethodForm) = {(BC, EmBody)}
δ(ATT, >, Em) = {(C, Em)}
δ(BC, , EmBody) = {(BC, e)}
δ(BC, , EmDiv) = {(BC, e)}
δ(BC, , EmForm) = {(BC, e)}
δ(BC, , EmMethodForm) = {(BC, e)}
δ(C, , Em) = {(C, e)}
δ(BC, <b, Body) = {(ATT, BBodyBody)}
δ(C, <b, P) = {(ATT, B)}
δ(C, <b, H1) = {(ATT, BH1)}
δ(C, <b, H2) = {(ATT, BH2)}
δ(C, <b, H3) = {(ATT, BH3)}
δ(C, <b, H4) = {(ATT, BH4)}
δ(C, <b, H5) = {(ATT, BH5)}
δ(C, <b, H6) = {(ATT, BH6)}
δ(C, <b, A) = {(ATT, BA)}
δ(C, <b, button) = {(ATT, BButton)}
δ(BC, <b, Div) = {(ATT, BDivDiv)}
δ(BC, <b, Form) = {(ATT, BFormForm)}
δ(BC, <b, MethodForm) = {(ATT, BMethodFormMethodForm)}
δ(ATT, >, BBody) = {(BC, BBody)}
δ(ATT, >, BDiv) = {(BC, BDiv)}
δ(ATT, >, BForm) = {(BC, BForm)}
δ(ATT, >, BMethodForm) = {(BC, BMethodForm)}
δ(ATT, >, B) = {(C, B)}
δ(BC, , BBody) = {(BC, e)}
δ(BC, , BDiv) = {(BC, e)}
δ(BC, , BForm) = {(BC, e)}
δ(BC, , BMethodForm) = {(BC, e)}
δ(C, , B) = {(C, e)}
δ(BC, <abbr, Body) = {(ATT, AbbrBodyBody)}
δ(C, <abbr, P) = {(ATT, Abbr)}
δ(C, <abbr, H1) = {(ATT, AbbrH1)}
δ(C, <abbr, H2) = {(ATT, AbbrH2)}
δ(C, <abbr, H3) = {(ATT, AbbrH3)}
δ(C, <abbr, H4) = {(ATT, AbbrH4)}

δ(C, <abbr, H5) = {(ATT, AbbrH5)}
δ(C, <abbr, H6) = {(ATT, AbbrH6)}
δ(C, <abbr, A) = {(ATT, AbbrA)}
δ(C, <abbr, button) = {(ATT, AbbrButton)}
δ(BC, <abbr, Div) = {(ATT, AbbrDivDiv)}
δ(BC, <abbr, Form) = {(ATT, AbbrFormForm)}
δ(BC, <abbr, MethodForm) = {(ATT,
AbbrMethodFormMethodForm)}
δ(ATT, >, AbbrBody) = {(BC, AbbrBody)}
δ(ATT, >, AbbrDiv) = {(BC, AbbrDiv)}
δ(ATT, >, AbbrForm) = {(BC, AbbrForm)}
δ(ATT, >, AbbrMethodForm) = {(BC, AbbrMethodForm)}
δ(ATT, >, Abbr) = {(C, Abbr)}
δ(BC, </abbr>, AbbrBody) = {(BC, e)}
δ(BC, </abbr>, AbbrDiv) = {(BC, e)}
δ(BC, </abbr>, AbbrForm) = {(BC, e)}
δ(BC, </abbr>, AbbrMethodForm) = {(BC, e)}
δ(C, </abbr>, Abbr) = {(C, e)}
δ(BC, <strong, Body) = {(ATT, StrongBodyBody)}
δ(C, <strong, P) = {(ATT, Strong)}
δ(C, <strong, H1) = {(ATT, StrongH1)}
δ(C, <strong, H2) = {(ATT, StrongH2)}
δ(C, <strong, H3) = {(ATT, StrongH3)}
δ(C, <strong, H4) = {(ATT, StrongH4)}
δ(C, <strong, H5) = {(ATT, StrongH5)}
δ(C, <strong, H6) = {(ATT, StrongH6)}
δ(C, <strong, A) = {(ATT, StrongA)}
δ(C, <strong, button) = {(ATT, StrongButton)}
δ(BC, <strong, Div) = {(ATT, StrongDivDiv)}
δ(BC, <strong, Form) = {(ATT, StrongFormForm)}
δ(BC, <strong, MethodForm) = {(ATT,
StrongMethodFormMethodForm)}
δ(ATT, >, StrongBody) = {(BC, StrongBody)}
δ(ATT, >, StrongDiv) = {(BC, StrongDiv)}
δ(ATT, >, StrongForm) = {(BC, StrongForm)}
δ(ATT, >, StrongMethodForm) = {(BC, StrongMethodForm)}
δ(ATT, >, Strong) = {(C, Strong)}
δ(BC, , StrongBody) = {(BC, e)}
δ(BC, , StrongDiv) = {(BC, e)}
δ(BC, , StrongForm) = {(BC, e)}
δ(BC, , StrongMethodForm) = {(BC, e)}

δ(C, , Strong) = {(C, e)}
δ(BC, <small, Body) = {(ATT, SmallBodyBody)}
δ(C, <small, P) = {(ATT, Small)}
δ(C, <small, H1) = {(ATT, SmallH1)}
δ(C, <small, H2) = {(ATT, SmallH2)}
δ(C, <small, H3) = {(ATT, SmallH3)}
δ(C, <small, H4) = {(ATT, SmallH4)}
δ(C, <small, H5) = {(ATT, SmallH5)}
δ(C, <small, H6) = {(ATT, SmallH6)}
δ(C, <small, A) = {(ATT, SmallA)}
δ(C, <small, button) = {(ATT, SmallButton)}
δ(BC, <small, Div) = {(ATT, SmallDivDiv)}
δ(BC, <small, Form) = {(ATT, SmallFormForm)}
δ(BC, <small, MethodForm) = {(ATT,
SmallMethodFormMethodForm)}
δ(ATT, >, SmallBody) = {(BC, SmallBody)}
δ(ATT, >, SmallDiv) = {(BC, SmallDiv)}
δ(ATT, >, SmallForm) = {(BC, SmallForm)}
δ(ATT, >, SmallMethodForm) = {(BC, SmallMethodForm)}
δ(ATT, >, Small) = {(C, Small)}
δ(BC, </small>, SmallBody) = {(BC, e)}
δ(BC, </small>, SmallDiv) = {(BC, e)}
δ(BC, </small>, SmallForm) = {(BC, e)}
δ(BC, </small>, SmallMethodForm) = {(BC, e)}
δ(C, </small>, Small) = {(C, e)}
δ(BC, <hr, Body) = {(ATT, HrBodyBody)}
δ(C, <hr, P) = {(ATT, Hr)}
δ(C, <hr, H1) = {(ATT, HrH1)}
δ(C, <hr, H2) = {(ATT, HrH2)}
δ(C, <hr, H3) = {(ATT, HrH3)}
δ(C, <hr, H4) = {(ATT, HrH4)}
δ(C, <hr, H5) = {(ATT, HrH5)}
δ(C, <hr, H6) = {(ATT, HrH6)}
δ(C, <hr, A) = {(ATT, HrA)}
δ(C, <hr, button) = {(ATT, HrButton)}
δ(BC, <hr, Div) = {(ATT, HrDivDiv)}
δ(BC, <hr, Form) = {(ATT, HrFormForm)}
δ(BC, <hr, MethodForm) = {(ATT, HrMethodFormMethodForm)}
δ(ATT, >, HrBody) = {(BC, e)}
δ(ATT, >, HrDiv) = {(BC, e)}
δ(ATT, >, HrForm) = {(BC, e)}

δ(ATT, >, HrMethodForm) = {(BC, e)}
δ(ATT, >, Hr) = {(C, e)}
δ(BC, <div, Body) = {(ATT, DivBody)}
δ(BC, <div, Div) = {(ATT, DivDiv+)}
δ(BC, <div, Form) = {(ATT, DivForm)}
δ(BC, <div, MethodForm) = {(ATT, DivMethodForm)}
δ(ATT, >, Div) = {(BC, Div)}
δ(BC, , Div) = {(BC, e)}
δ(BC, <a, Body) = {(ATT, ABody)}
δ(BC, <a, Div) = {(ATT, ADiv)}
δ(BC, <a, Form) = {(ATT, AForm)}
δ(BC, <a, MethodForm) = {(ATT, AMethodForm)}
δ(ATT, href=", A) = {(ATTCONTENT, HypertextReferenceA)}
δ(ATT, >, A) = {(C, A)}
δ(C, , A) = {(BC, e)}
δ(BC, <img, Body) = {(ATT, ImgBody)}
δ(BC, <img, Div) = {(ATT, ImgDiv)}
δ(BC, <img, Form) = {(ATT, ImgForm)}
δ(BC, <img, MethodForm) = {(ATT, ImgMethodForm)}
δ(ATT, src=", Img) = {(ATTCONTENT, SourceSourceImg)}
δ(ATT, alt=", Img) = {(ATTCONTENT, AlternativeImg)}
δ(ATT, alt=", SourceImg) = {(ATTCONTENT,
AlternativeSourceImg)}
δ(ATT, >, SourceImg) = {(BC, e)}
δ(BC, <button, Body) = {(ATT, ButtonBody)}
δ(BC, <button, Div) = {(ATT, ButtonDiv)}
δ(BC, <button, Form) = {(ATT, ButtonForm)}
δ(BC, <button, MethodForm) = {(ATT, ButtonMethodForm)}
δ(ATT, type=", Button) = {(ATTBUTTON, TypeButton)}
δ(ATTBUTTON, submit", Type) = {(ATT, e)}
δ(ATTBUTTON, reset", Type) = {(ATT, e)}
δ(ATTBUTTON, button", Type) = {(ATT, e)}
δ(ATT, >, Button) = {(C, Button)}
δ(C, </button>, Button) = {(BC, e)}
δ(BC, <form, Body) = {(ATT, FormBody)}
δ(BC, <form, Div) = {(ATT, FormDiv)}
δ(ATT, action=", Form) = {(ATTCONTENT, ActionForm)}
δ(ATT, action=", MethodForm) = {(ATTCONTENT,
ActionMethodForm)}
δ(ATT, method=", Form) = {(ATTFORM, Form)}
δ(ATTFORM, get", Form) = {(ATT, MethodForm)}

δ(ATTFORM, post", Form) = {(ATT, MethodForm)}
δ(ATT, >, Form) = {(BC, Form)}
δ(ATT, >, MethodForm) = {(BC, MethodForm)}
δ(BC, </form>, Form) = {(BC, e)}
δ(BC, </form>, MethodForm) = {(BC, e)}
δ(BC, <input, Body) = {(ATT, InputBody)}
δ(BC, <input, Div) = {(ATT, InputDiv)}
δ(BC, <input, Form) = {(ATT, InputForm)}
δ(BC, <input, MethodForm) = {(ATT, InputMethodForm)}
δ(ATT, type=", Input) = {(ATTINPUT, Input)}
δ(ATTINPUT, text", Input) = {(ATT, TypeInput)}
δ(ATTINPUT, password", Input) = {(ATT, TypeInput)}
δ(ATTINPUT, email", Input) = {(ATT, TypeInput)}
δ(ATTINPUT, number", Input) = {(ATT, TypeInput)}
δ(ATTINPUT, checkbox", Input) = {(ATT, TypeInput)}
δ(ATT, >, TypeInput) = {(BC, e)}δ(ATT, >, Input) = {(BC, e)}
δ(BC, <table, Body) = {(ATT, TableBody)}
δ(BC, <table, Div) = {(ATT, TableDiv)}
δ(BC, <table, Form) = {(ATT, TableForm)}
δ(BC, <table, MethodForm) = {(ATT, TableMethodForm)}
δ(ATT, >, Table) = {(TC, Table)}
δ(TC, </table>, Table) = {(BC, e)}
δ(TC, <tr, Table) = {(ATT, TableRowTable)}
δ(ATT, >, TableRow) = {(TC, TableRow)}
δ(TC, </tr>, TableRow) = {(TC, e)}
δ(TC, <td, TableRow) = {(ATT, TableDataTableRow)}
δ(ATT, >, TableData) = {(C, TableData)}
δ(C, </td>, TableData) = {(TC, e)}
δ(TC, <th, TableRow) = {(ATT, TableHeaderTableRow)}
δ(ATT, >, TableHeader) = {(C, TableHeader)}
δ(C, </th>, TableHeader) = {(TC, e)}

3.5. Start State

Q

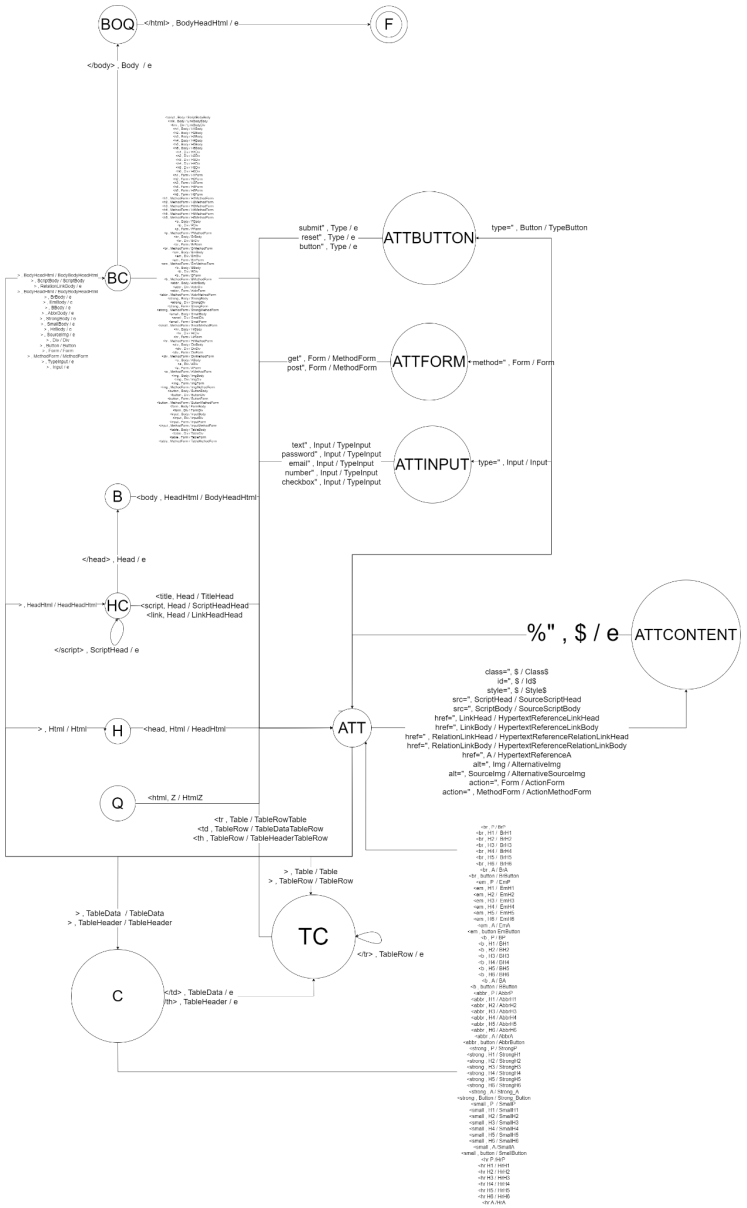
3.6. Start Stack

Z

3.7. Final States

F

3.8. Diagram PDA



Untuk gambar yang lebih jelasnya dapat dilihat di [sini](https://bit.ly/DiagramPDA) (https://bit.ly/DiagramPDA)

Bab IV

Implementasi dan Pengujian

4.1. Implementasi

Pada tugas besar Tata Bahasa Formal Automata ini kami menerapkan pushdown automata dengan bahasa pemrograman Python. Program ini menggunakan antarmuka berbasis CLI (Command Line Interface) . Program akan menerima suatu teks file yang merupakan kode HTML, lalu program akan mengolahnya dengan memanfaatkan PDA untuk memvalidasi kebenaran syntax dari kode input.

4.1.1. HTML Reader :

Module ini digunakan untuk membaca file html dan mengubahnya menjadi token yang disetor ke dalam sebuah list. Di dalam module ini terdapat dua fungsi, yaitu get_raw_list dan get_remainding_input.

a. get_raw_list(file_html) : Menerima path ke file html dan menghasilkan sebauh list yang berisi kata per kata dari file html tersebut.
b. get_remainding_input(file_html) : Menerima path ke file html dan memanggil get_raw_list(file_html). Setelah itu, hasil dari get_raw_list akan diolah sehingga menjadi token yang diterima progeram.

4.1.2. PDAReader :

Module ini digunakan untuk membaca file PDA menjadi beberapa variable untuk menyimpan data PDA yaitu states, input dan stack alphabet, start state dan stack, final states, dan transition functions.

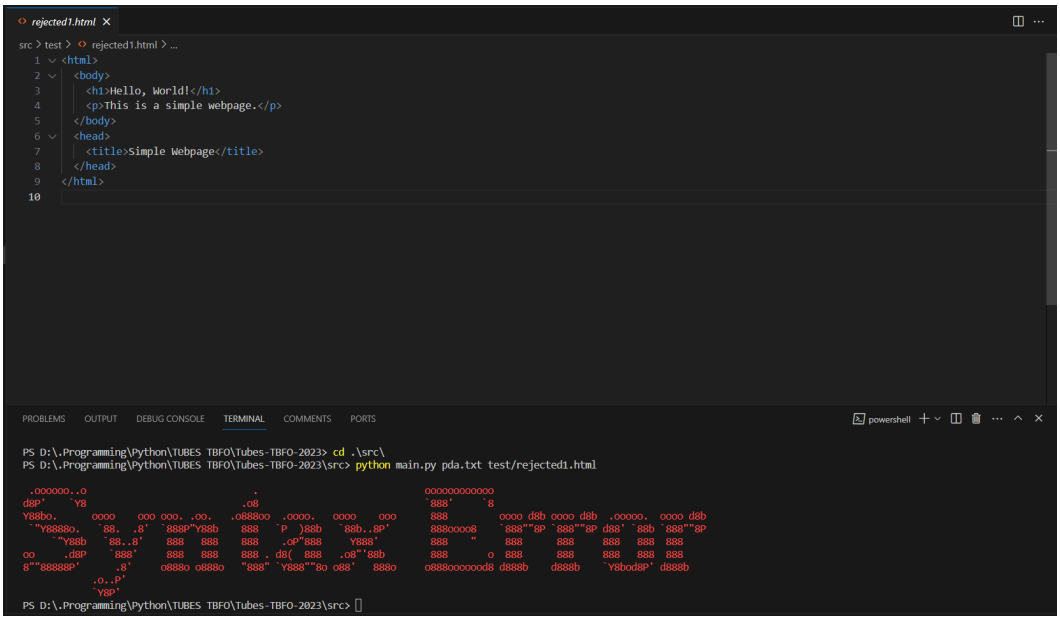
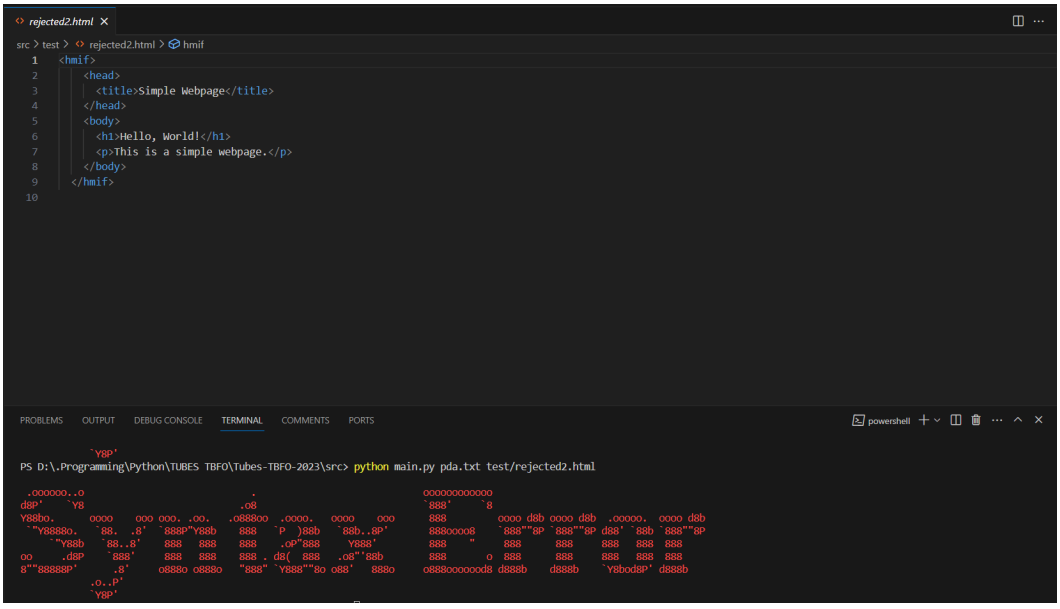
4.1.3 PDA:

File PDA merupakan file text document yang diperlukan untuk menerima peraturan HTML Checker dan sebagai panduan untuk aplikasi stack pada program main. Pada file ini berisi sejumlah state, input, stack, dan lain-lain.

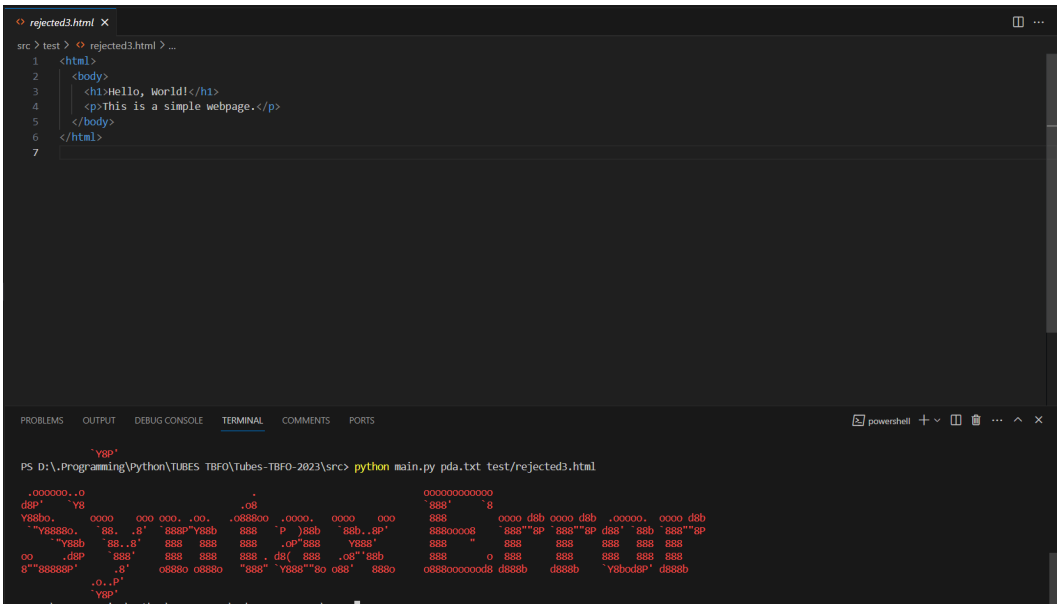
4.1.4 Main:

Program main ini menerima masukan command di terminal berupa file PDA dan HTML untuk dicek kebenarannya. File ini mengintegrasikan HTML Reader, PDAReader dan file PDA sehingga program dapat berjalan.

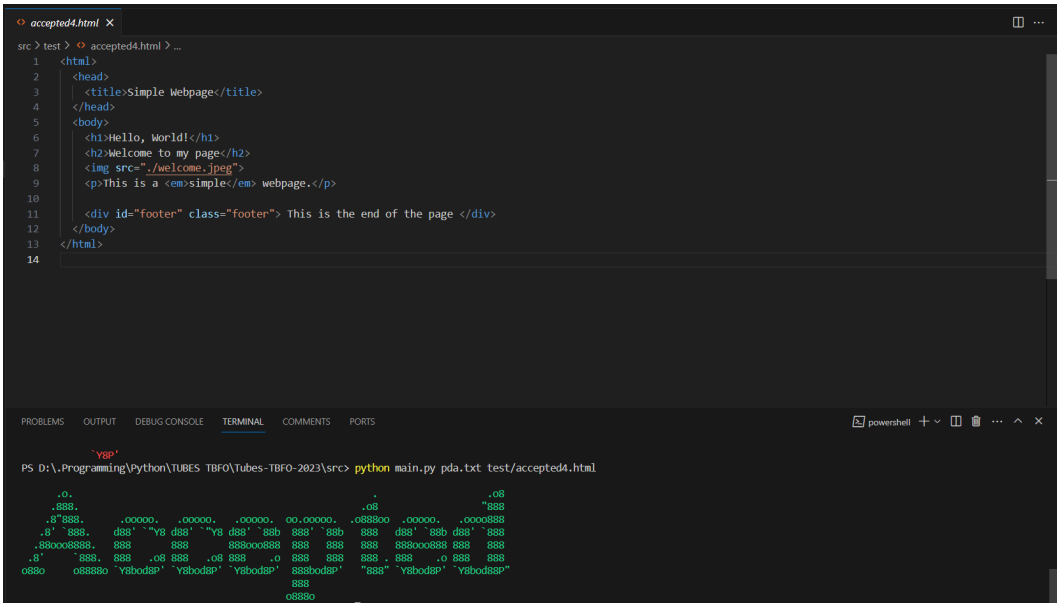
4.2. Testing

No	Hasil
1.	
2.	

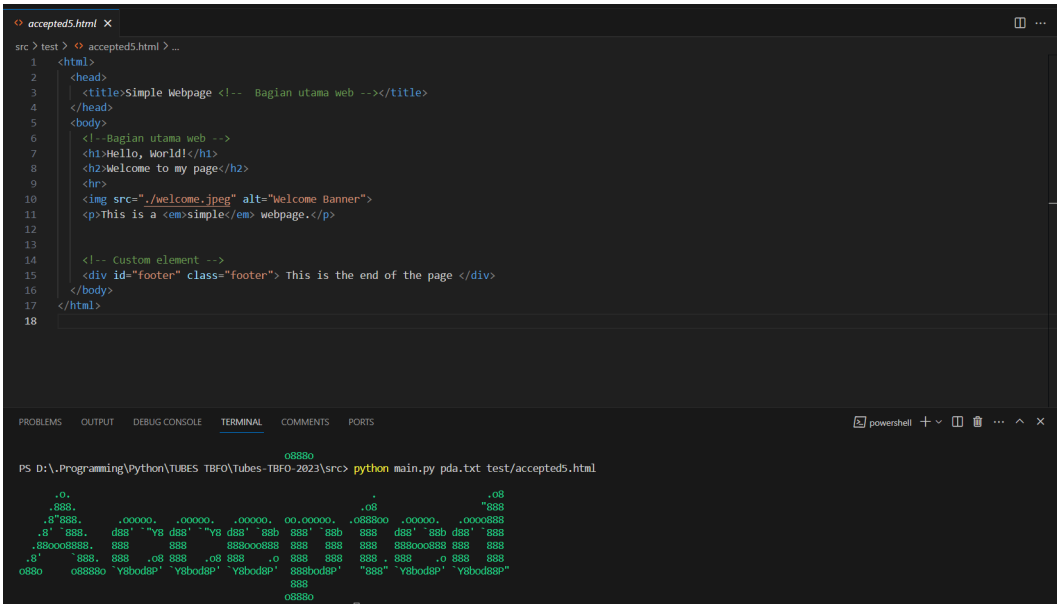
3.


rejected3.html
src > test > rejected3.html > ...
1 <html>
2 <body>
3 <h1>Hello, World!</h1>
4 This is a simple webpage.
5 </body>
6 </html>
7
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
powershell
`Y8P'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/rejected3.html

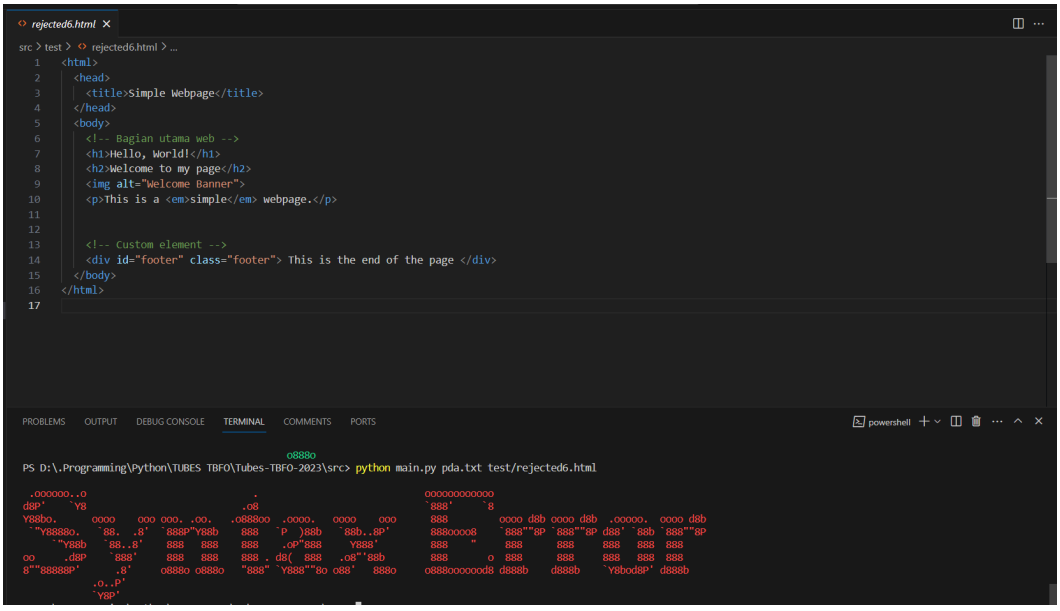
4.


accepted4.html
src > test > accepted4.html > ...
1 <html>
2 <head>
3 <title>Simple Webpage</title>
4 </head>
5 <body>
6 <h1>Hello, World!</h1>
7 <h2>Welcome to my page</h2>
8 <img src="./welcome.jpeg">
9 This is a simple webpage.
10
11 This is the end of the page
12 </body>
13 </html>
14
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
powershell
`Y8P'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/accepted4.html

5.


accepted5.html
src > test > accepted5.html > ...
<html>
<head>
<title>Simple Webpage <!-- Bagian utama web --></title>
</head>
<body>
<!--Bagian utama web -->
<h1>Hello, World!</h1>
<h2>Welcome to my page</h2>
<hr>
<img src="./welcome.jpeg" alt="Welcome Banner">
This is a simple webpage.
<!-- Custom element -->
 This is the end of the page
</body>
</html>
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
powershell
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/accepted5.html

6.


rejected6.html
src > test > rejected6.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">
This is a simple webpage.
<!-- Custom element -->
 This is the end of the page
</body>
</html>
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
powershell
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/rejected6.html

7.

accepted7.html

```
src > test > accepted7.html > html
1  <html>
2  <head>
3    <title>Simple Webpage</title>
4
5  </head>
6  <body>
7
8  <h2>HTML Forms</h2>
9
10 <form action="/action_page.php" method="POST"> <!-- komenr-->
11   <h5     class="label">First name:</h5><br>
12   <input type="text" id="fname"><br>
13   <h5 class="label">Last name:</h5><br>
14   <input type="text" id="lname"><br><br>
15   <button type="submit">check</button>
16 </form>
17
18 <p>If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".</p>
19
20 </body>
21 </html>
22
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS — powershell

```
            `Y8P'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/accepted7.html

      .o.                                                .                  .o8
     .888.                                             .o8                 "888
    .8"888.      .ooooo.   .ooooo.   .ooooo.  oo.ooooo.  .o888oo  .ooooo.   .oooo888
   .8' `888.    d88' `"Y8 d88' `"Y8 d88' `88b  888' `88b   888   d88' `88b d88' `888
  .88ooo8888.   888       888       888ooo888  888   888   888   888ooo888 888   888
 .8'     `888.  888   .o8 888   .o8 888    .o  888   888   888 . 888    .o 888   888
o88o     o8888o `Y8bod8P' `Y8bod8P' `Y8bod8P'  888bod8P'   "888" `Y8bod8P' `Y8bod88P"
                                               888
                                               o888o
```

8.

rejected8.html

```
src > test > rejected8.html > html
1  <html>
2  <head>
3    <title>Simple Webpage</title>
4
5  </head>
6  <body>
7
8  <h2>HTML Forms</h2>
9
10 <form action="/action_page.php" method="TEMBAK">
11   <div id="label">First name:</div><br>
12   <input type="text" id="fname"><br>
13   <div id="label">Last name:</div><br>
14   <input type="text" id="lname"><br><br>
15   <button type="submit">Submit</button>
16 </form>
17
18 <p>If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".</p>
19
20 </body>
21 </html>
22
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS — powershell

```
FileNotFoundError: [Errno 2] No such file or directory: 'test/accepted8.html'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/rejected8.html

 .oooooo..o                                .                                oooooooooooo
d8P'    `Y8                              .o8                                `888'     `8
Y88bo.      oooo    ooo ooo. .oo.   .o888oo  .oooo.    oooo    ooo         888          oooo d8b oooo d8b  .ooooo.  oooo d8b
 `"Y8888o.   `88.  .8' `888P"Y88b    888    `P  )88b    `88b..8P'          888oooo8     `888""8P `888""8P d88' `88b `888""8P
     `"Y88b   `88..8'   888   888    888     .oP"888      Y888'            888     "      888      888     888   888  888
oo     .d8P    `888'    888   888    888 . d8(  888    .o8"'88b            888       o    888      888     888   888  888
8""88888P'      .8'     o888o o888o   "888" `Y888""8o o888'   888o         o888oooooood8 d888b    d888b    `Y8bod8P' d888b
             .o..P'
             `Y8P'
```

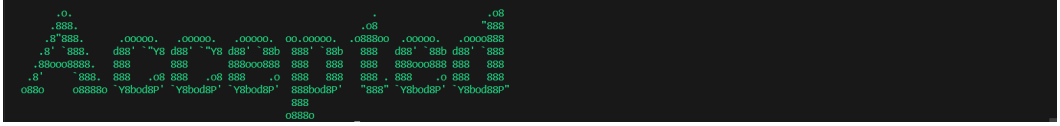
9.

accepted9.html

```
src > test > accepted9.html > ...
1  <html>
2      <head>
3          <title>Simple Webpage</title>
4          <script>
5              document.getElementById("demo").innerHTML = "Hello JavaScript!";
6          </script>
7      </head>
8      <body>
9          <h1>The script element</h1>
10         <a>Not going anywhere</a><br>
11         <a href="https://www.google.co.id/">Might send you somewhere</a>
12
13         <p id="demo"></p>
14     </body>
15 </html>
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS — powershell

```
FileNotFoundError: [Errno 2] No such file or directory: 'test/rejected9.html'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/accepted9.html
```



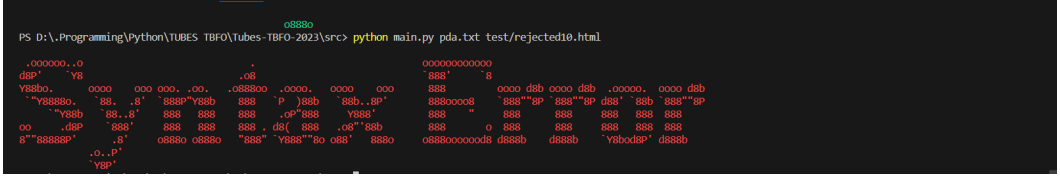
10.

rejected10.html

```
src > test > rejected10.html > html
1  <html>
2  <head>
3    <title>Simple Webpage</title>
4    <script>
5      document.getElementById("demo").innerHTML = "Hello JavaScript!";
6    </script>
7  </head>
8  <body>
9
10 <h1>The script element</h1>
11
12 <p id="demo">
13
14 </body>
15 </html>
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS — powershell

```
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/rejected10.html
```



11.

accepted10.html

```
src > test > accepted10.html > ...
1  <html>
2      <head>
3          <title>Simple Webpage</title>
4          <script>
5              document.getElementById("demo").innerHTML = "Hello JavaScript!";
6          </script>
7      </head>
8      <body>
9          <h1>The script element</h1>
10         <a>Not going anywhere</a><br>
11         <a href="https://www.google.co.id/">Might send you somewhere</a>
12
13         <p id="demo"></p>
14     </body>
15 </html>
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS — powershell

```
            `Y8P'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/accepted10.html

      .o.                                                .                    .o8
     .888.                                             .o8                   "888
    .8"888.      .ooooo.   .ooooo.   .ooooo.  oo.ooooo.  .o888oo  .ooooo.    .oooo888
   .8' `888.    d88' `"Y8 d88' `"Y8 d88' `88b  888' `88b   888    d88' `88b d88' `888
  .88ooo8888.   888       888       888ooo888  888   888   888    888ooo888 888   888
 .8'     `888.  888   .o8 888   .o8 888     .o  888   888   888 .  888     .o 888   888
o88o     o8888o `Y8bod8P' `Y8bod8P' `Y8bod8P'  888bod8P'   "888" `Y8bod8P' `Y8bod88P"
                                               888
                                              o888o
```

12.

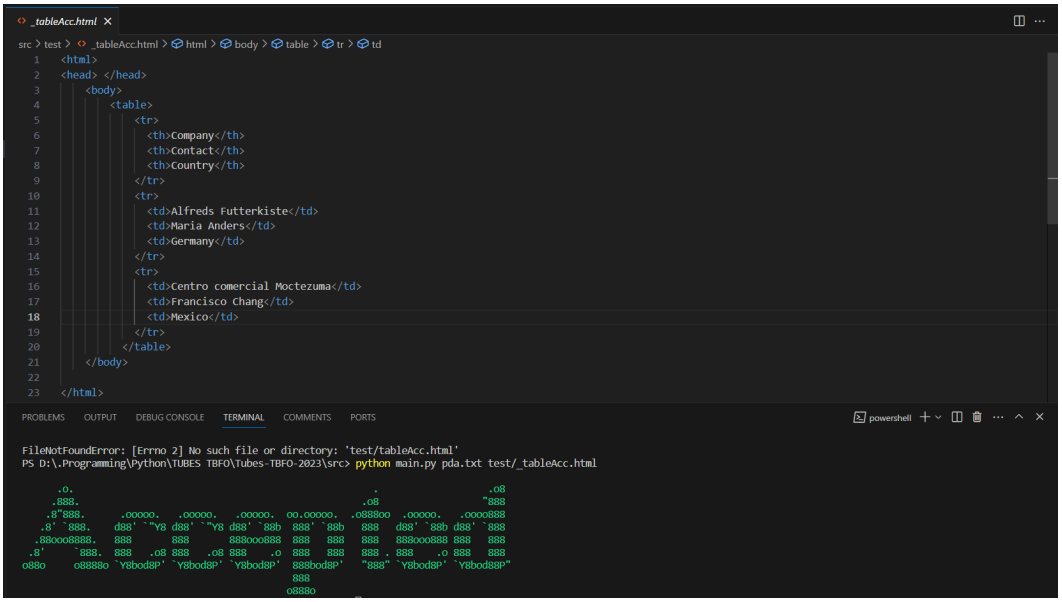
_linkAcc.html

```
src > test > _linkAcc.html > html
1  <html>
2
3  <head>
4      <title>Valid Link Example</title>
5  </head>
6
7  <body>
8      <link rel="stylesheet" href="styles.css">
9      <h2>Content Goes Here</h2>
10     <p>This HTML document includes a valid  tag with the "stylesheet" relationship.</p>
11
12 </body>
13
14 </html>
15
```

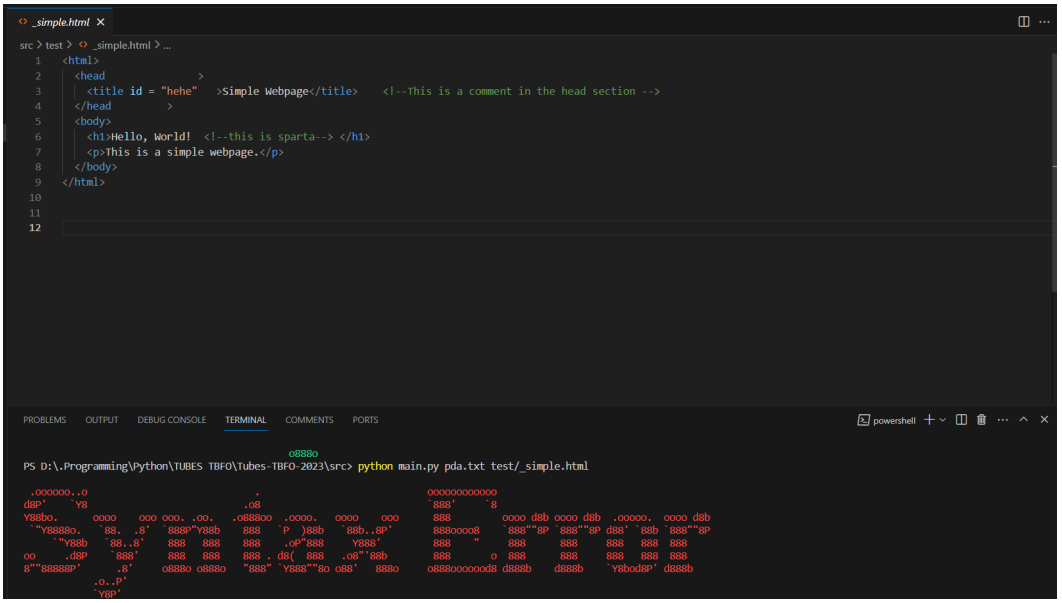
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS — powershell

```
                                              o888o
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/_linkAcc.html

      .o.                                                .                    .o8
     .888.                                             .o8                   "888
    .8"888.      .ooooo.   .ooooo.   .ooooo.  oo.ooooo.  .o888oo  .ooooo.    .oooo888
   .8' `888.    d88' `"Y8 d88' `"Y8 d88' `88b  888' `88b   888    d88' `88b d88' `888
  .88ooo8888.   888       888       888ooo888  888   888   888    888ooo888 888   888
 .8'     `888.  888   .o8 888   .o8 888     .o  888   888   888 .  888     .o 888   888
o88o     o8888o `Y8bod8P' `Y8bod8P' `Y8bod8P'  888bod8P'   "888" `Y8bod8P' `Y8bod88P"
                                               888
                                              o888o
```

13.


_tableAcc.html
src > test > _tableAcc.html > html > body > table > tr > td
<html>
<head> </head>
<body>
<table>
<tr>
<th>Company</th>
<th>Contact</th>
<th>Country</th>
</tr>
<tr>
<td>Alfreds Futterkiste</td>
<td>Maria Anders</td>
<td>Germany</td>
</tr>
<tr>
<td>Centro comercial Moctezuma</td>
<td>Francisco Chang</td>
<td>Mexico</td>
</tr>
</table>
</body>
</html>
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
powershell
FileNotFoundError: [Errno 2] No such file or directory: 'test/tableAcc.html'
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/_tableAcc.html

14.


_simple.html
src > test > _simple.html > ...
<html>
<head >
<title id = "hehe" >Simple Webpage</title> <!--This is a comment in the head section -->
</head >
<body>
<h1>Hello, World! <!--this is sparta--> </h1>
This is a simple webpage.
</body>
</html>
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
powershell
PS D:\.Programming\Python\TUBES TBFO\Tubes-TBFO-2023\src> python main.py pda.txt test/_simple.html

15.

testcase.html U | HTMLreader.py | pda.txt | PDAReader.py | main.py | ini.txt U

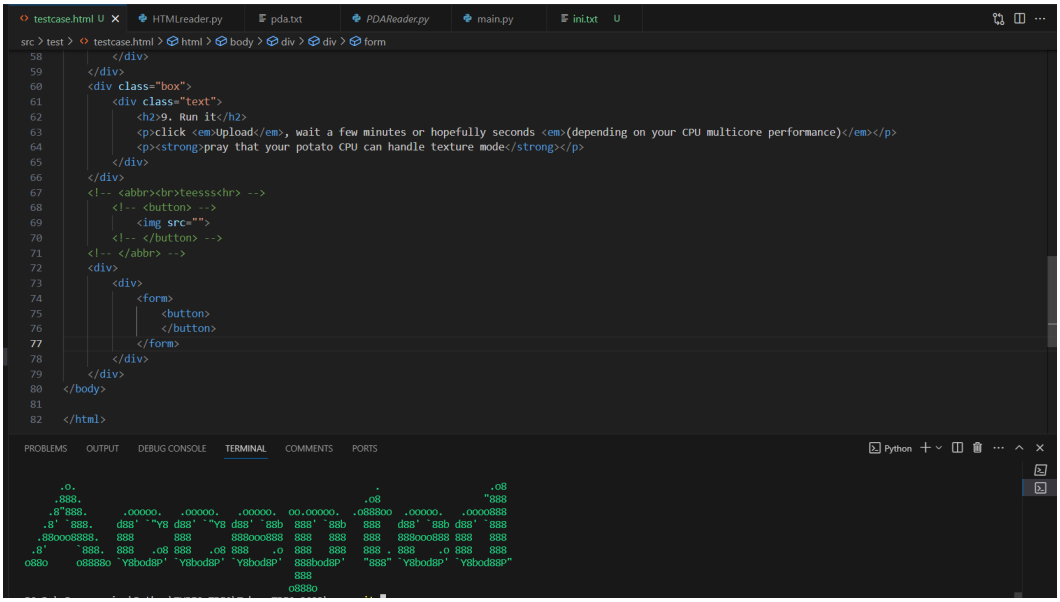
src > test > testcase.html > html > body > div > div > form

```
<html>

<head>
    <title>How To Use</title>
    <link rel="stylesheet" href="../static/aboutus_style.css">
</head>

<body>
    <div id="title">
        <h1>How To Use</h1>
    </div>
    <div class="box">
        <div class="text">
            <h2>1. Clone the repo</h2>
            <p>This can be done by opening up the terminal, change directory to the desired location, and type: git clone git@github.com:riyorax/Algeo02-22061.gi
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>2. Navigate to the repo's source file directory</h2>
            <p>Change directory to the cloned git repo by typing: cd ./Algeo02-22061/src/</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>3. Install the dependencies</h2>
            <p>Change directory to the cloned git repo and type: pip install -r requirements.txt or python3 -m pip install -r requirements.txt</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>4. Run the website locally</h2>
            <p>This is done by typing: py app.py or python3 app.py</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
```

testcase.html U | HTMLreader.py | pda.txt | PDAReader.py | main.py | ini.txt U

src > test > testcase.html > html > body > div > div > form

```
    <div class="box">
        <div class="text">
            <h2>5. Open the localhost website</h2>
            <p>Look at the terminal output, see where the web is hosted at and Crtl + click</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>6. Click the toggle to switch between color or texture mode</h2>
            <p>Just click it, as simple as that</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>7. <em>Upload Image</em> section</h2>
            <p>Just click <em>Choose File</em>, select <em>one</em> image</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>8. <em>Upload Dataset</em> section</h2>
            <p>Just click <em>Choose Files</em>, select one or more images or even folders</p>
        </div>
    </div>
    <div class="box">
        <div class="text">
            <h2>9. Run it</h2>
            <p>click <em>Upload</em>, wait a few minutes or hopefully seconds <em>(depending on your CPU multicore performance)</em></p>
            <p><strong>pray that your potato CPU can handle texture mode</strong></p>
        </div>
    </div>
    <!-- <abbr><br>teesss<hr> -->
        <!-- <button> -->
            <img src="">
        <!-- </button> -->
    <!-- </abbr> -->
    <div>
```


testcase.html U
HTMLreader.py
pda.txt
PDAReader.py
main.py
ini.txt U
src > test > testcase.html > html > body > div > div > div > form

<h2>9. Run it</h2>
click Upload, wait a few minutes or hopefully seconds (depending on your CPU multicore performance)
pray that your potato CPU can handle texture mode

<!-- <abbr>
teesss<hr> -->
<!-- <button> -->
<img src="">
<!-- </button> -->
<!-- </abbr> -->

<form>
<button>
</button>
</form>

</body>
</html>
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
COMMENTS
PORTS
Python

4.3.

Bab V

Kesimpulan

5.1 Kesimpulan

Melalui tugas besar Tata Bahasa Formal Automata kami dapat mempelajari Pushdown Automata lebih jauh lagi dan aplikasinya sebagai HTML checker. Selain belajar PDA kami juga meningkatkan skill programming kami lebih lagi khususnya di python dan HTML. Bukan hanya kemampuan hard skill yang kami pelajari kami juga menyadari pentingnya komunikasi dalam melakukan tugas berkelompok untuk menghindari bug bug yang perlu didebugging. Selain itu kami juga meningkatkan rasa persahabatan ketika melakukan tugas TBFO ini. Dibalik siapnya tugas besar ini kami juga masing-masing melakukan kesalahan tersendiri sehingga harus melakukan refleksi lebih banyak dan mengoreksinya demi hasil yang lebih maksimal kedepannya.

5.2 Pembagian Tugas

No	Nama	NIM	Deskripsi
1	Devinzen	13522064	Mengimplementasikan PDA, laporan
2	Matthew Vladimir Hutabarat	13522093	Membuat file dan diagram PDA, laporan
3	Berto Richardo	13522118	Membuat parser HTML, laporan

	Togatorop		

Lampiran

Link repository github

https://github.com/NgokNgok04/Algeo02-22064

Link diagram state

https://drive.google.com/file/d/1zISvnmJIUDWcTEu_dRr_V8ztu4WNvqTs/view?usp=sharing