**LAPORAN PROYEK TEORI BAHASA FORMAL DAN OTOMATA**

**INTERAKTIF VISUALISASI: FINITE STATE AUTOMATA**

**DAN MESIN TURING DENGAN JAVASCRIPT**



**Dosen Pengampu :**

**Nurul Hayaty, S.T., M.Cs**

**Disusun Oleh :**

**Kelompok 10**

Fauzan Aldi 2301020022

Hermawan 2301020025

Raja Aizin Nofrizivaldy 2301020027

Masry Ryzki Yanditar 2301020087

Roy Adiyta 2301020093

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN TEKNOLOGI KEMARITIMAN**

**UNIVERSITAS MARITIM RAJA ALI HAJI**

**TANJUNGPINANG**

**2024**

**DAFTAR ISI**

**BAB I**

**PENDAHULUAN**

**1.1 Latar Belakang**

Pemahaman tentang Mesin Turing sering menjadi tantangan, terutama bagi mahasiswa dan peneliti yang baru mempelajari teori komputasi. Penyampaian materi yang hanya menggunakan pendekatan matematis atau teks tidak cukup untuk menjelaskan bagaimana Mesin Turing bekerja dalam praktiknya, seperti transisi antar state atau manipulasi pita. Hal ini menciptakan kebutuhan akan alat bantu visual yang dapat menggambarkan operasional Mesin Turing secara interaktif.

Visualisasi interaktif berbasis web adalah salah satu pendekatan yang dapat menjawab tantangan ini. Dengan memanfaatkan bahasa pemrograman seperti JavaScript, struktur web HTML5, dan tampilan dengan CSS, simulasi Mesin Turing dapat divualisasikan dalam format yang mudah diakses melalui perangkat misalnya dengan browser web. Visualisasi ini memungkinkan pengguna untuk memahami cara kerja dari mesin turing dalam membaca tape.

Laporan ini dibuat untuk melengkapi tugas proyek akhir mata kuliah Teori Bahasa Formal dan otomata dengan judul “Interaktif Visualisasi: Finite State Automata dan Mesin Turing dengan JavaScript”.

**1.2 Rumusan Masalah**

Adapun beberapa rumusan masalah pada laporan ini yaitu, sebagai berikut:

1. Bagaimana membuat *website* interaktif yang dapat memvisualisasikan konsep abstrak FSA dan Mesin Turing?
2. Bagaimana memahami cara kerja mesin turing dan mengimplementasikannya ke dalam sebuah *website*?

**1.3 Tujuan Penulisan**

Adapun beberapa tujuan pada laporan ini yaitu, sebagai berikut:

1. Membuat dan memahami proses mengimplementasikan konsep FSA dan mesin turing ke dalam sebuah *website* agar dapat divisulisasikan.
2. Mengindentifikasikan dan membangun logika pemrograman yang bekerja layaknya mesin turing pada bahasa *javascript*.

**1.4 Manfaat Penulisan**

Adapun manfaat penulisan laporan ini adalah agar dapat memahami lebih dalam konsep mesin turing dan juga dapat melihat bagaimana mesin turing bekerja serta berinteraksi langsung pada pergerakan diagram FSA.

**1.5. Landasan Teori**

Adapun beberapa landasan teori yang menjadi referensi pada laporan ini yaitu, sebagai berikut:

**1.5.1 Mesin Turing**

Mesin turing pada awalnya diusulkan oleh seorang matematikawan inggris bernama Alan Turing pada tahun 1936. Alan Turing lahir pada 23 Juni 1912 di Maide Vale, London. Ia dikenal sebagai bapak dari ilmu komputer dan kecerdasan buatan. Alan Turing memperkenalkan konsep mesin turing dalam rangka penelitian tentang dasar-dasar matematika. Secara khusus, ia menggunakan konsep abstrak ini untuk menyelesaikan permasalah *Entscheidungsproblem*.

Masalah Entscheidungsproblem merupakan masalah untuk menentukan, pada setiap pernyataan dalam logika orde pertama (*first-order logic*), apakah pernyataan tersebut dapat diturunkan atau tidak dalam logika tersebut.

Pada bentuk awalnya (Hilbert & Ackermann, 1928), masalah ini dinyatakan dalam istilah validitas, bukan keterurutan (*derivability*). Namun, berdasarkan Teorema Kelengkapan Gödel (Gödel, 1929), pembuktian adanya prosedur efektif untuk keterurutan juga menyelesaikan masalah ini dalam bentuk validitasnya. Untuk menangani masalah ini, diperlukan konsep formal tentang "prosedur efektif," dan *Turing machines* dirancang untuk memenuhi kebutuhan tersebut.

Mesin turing adalah perangkat komputasi abstrak sederhana yang dimaksudkan untuk membantu menyelidiki sejauh mana dan keterbatasan apa yang dapat dihitung. 'Mesin otomatis' Turing, sebagaimana ia menyebutnya pada tahun 1936, secara khusus dirancang untuk menghitung bilangan riil. Mesin ini pertama kali diberi nama 'Mesin Turing' oleh Alonzo Church dalam tinjauan makalah Turing (Church 1937). Saat ini, mesin ini dianggap sebagai salah satu model dasar komputabilitas dan ilmu komputer (teoretis).

Mesin turing atau biasa disebut *turing machine* (atau *computing macine*, sebagaimana Turing menyebutnya) merupakan seperangkat konfigurasi (*states*) terbatas, yaitu *q1,q2,...,qn*. Mesin ini dilengkapi dengan pita satu dimensi dengan panjang tak terbatas, bergerak satu arah, yang dibagi menjadi kotak-kotak. Setiap kotak dapat menyimpan satu simbol yang bisa berupa kosong (dilambangkan *S0*) atau berisi simbol *S1*, *S2,...,Sm,* di mana *S1* = 0 dan *S1* = 1.

Terdapat beberapa tindakan mesin turing yaitu:

1. Mencetak *Si*, bergerak satu kotak ke kiri (L), dan masuk ke *state* *qj*.
2. Mencetak *Si*, bergerak satu kotak ke kanan (R), dan masuk ke *state qj*.
3. Mencetak *Si*, tetap di tempat (N), dan masuk ke *state qj*.

**Definisi Formal Mesin Turing**

Konsep “tape” dan “head” digunakan untuk mempermudah pemahaman intuitif. Dalam situasi yang membutuhkan analisis formal, definisi perangkat dan programnya lebih baik dinyatakan dalam istilah matematika.

Secara formal, sebuah mesin turing dapat didefinisikan sebagai berikut:

**T=(Q,Σ,** **Γ,δ,S,B,F)**

di mana:

* Q: himpunan dari *states*.
* Σ: himpunan dari simbol masukan.
* Γ: himpunan simbol pada *tape*.
* S: merupakan *state* awal.
* F: merupakan *state* akhir.
* B: simbol blank.
* δ: fungsi transisi yang menentukan langkah berikutnya, dengan bentuk:

**δ : (Q×Σ)→(Σ×{L,R}×Q)**

#### **Penjelasan Fungsi Transisi δ:**

Fungsi transisi ini menggambarkan bagaimana mesin bergerak dari satu kondisi ke kondisi lainnya.

**δ(*qi​,Sj*​)=(*Si,j​,D,qi,j*​)**

berarti bahwa:

1. Ketika mesin berada pada *state* *qi* dan membaca simbol *Sj*.
2. Mesin akan mengganti simbol *Sj* dengan *Si,j*.
3. Bergerak ke arah D∈{L,R} (kiri atau kanan),
4. Dan masuk ke *state* berikutnya *qi,j*​.

#### Contoh:

Misalnya, jika fungsi transisi diberikan sebagai berikut:

**δ(*q1​,S0​*)=(*S1​,R,q2****​*)

Maka artinya:

* Jika mesin berada pada *state* *q1* dan membaca simbol *S0*,
* Mesin mengganti *S0* dengan *S1*​,
* Bergerak ke kanan (R),
* Dan berpindah ke *state* *q2*.

**1.5.2 Finite State Automata**

Finite State Automata (FSA) adalah model matematika yang dapat menerima input dan output berupa data diskrit yang memiliki kemampuan untuk mengenali bahasa reguler. FSA memiliki sejumlah state yang terbatas, dengan kemampuan berpindah dari satu state ke state lainnya.

Finite Automata terdiri dari beberapa komponen utama. Karena memiliki sekumpulan state dan aturan untuk berpindah antar state, finite automata didefinisikan dengan lima elemen yang dikenal sebagai 5-tuple. Secara formal, finite automata dijelaskan sebagai kumpulan dari lima elemen ini, yaitu kumpulan state, simbol input, aturan transisi, state awal, dan state akhir. Kelima elemen tersebut dilambangkan dengan:

M = (Q, Σ, δ, S, F)

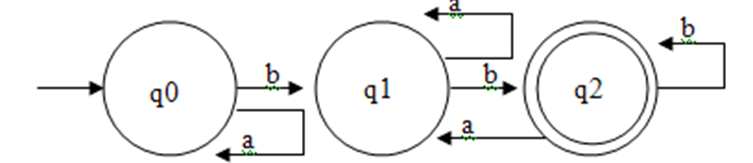
* Q: Himpunan state
* Σ: Himpunan simbol input
* δ: Fungsi transisi
* S: State awal (initial state)
* F: State akhir (final state)

FSA berdasarkan pada pendefinisian kemampuan berubah state-statenya bisa dikelompokkan menjadi dua yaitu DFA dan NFA:

**Deterministic Finite Automata (DFA)**

Pada DFA dari suatu state ada tepat satu state berikutnya untuk setiap simbol input (masukan) yang di terima.

Contoh:



Konfigurasi DFA contoh di atas secara formal adalah sebagai berikut :

Q = {q0, q1, q2}

Σ = {a, b}

S = q0

F = {q2}

Fungsi-fungsi transisinya sebagai berikut :

δ(q0, a) = q0, δ(q0, b) = q1,

δ(q1, a) = q1, δ(q1, b) = q2,

δ(q2, a) = q1, δ(q2, b) = q2.

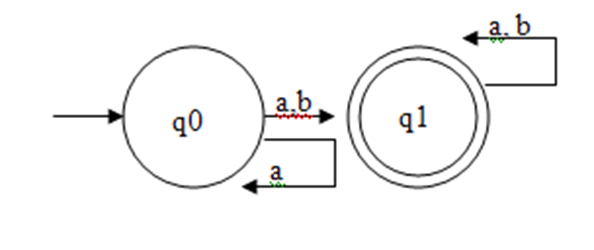
Jika disajikan dalam tabel transisi :

|  |  |  |
| --- | --- | --- |
| δ | a | b |
| q0 | q0 | q1 |
| q1 | q1 | q2 |
| q2 | q1 | q2 |

**Non Deterministic Finite Automata (NFA)**

Pada NFA dari suatu state bisa terdapat nol (0), satu (1), atau lebih busur keluar (transisis) berlabel simbol yang sama. Jadi setiap pasangan state-input bisa memiliki 0 atau lebih pilihan untuk state berikutnya.

Contoh:



Pada NFA contoh diatas terdapat dua masukan yang keluar berlabel input ‘a’. Dari state q0 bila mendapat input ‘a’ bisa berpindah ke state q0 atau q1 yang secara formal dinyatakan :

(q0, a) = {q0, q1}

Berikut definisi formal dari NFA di atas:

Q = {q0, q1 }

Σ = {a, b}

S = q0

F = {q1}

Fungsi-fungsi transisinya sebagai berikut :

(q0, a) = {q0,q1}, (q0, b) = q1,

(q1, a) = q1, (q1, b) = q1,

Jika disajikan dalam tabel transisi :

|  |  |  |
| --- | --- | --- |
| δ | a | b |
| q0 | {q0,q1} | q1 |
| q1 | q1 | q1 |

**1.5.3 Javascript**

JavaScript sering disingkat JS, adalah bahasa pemrograman tingkat tinggi yang bersifat dinamis. Bahasa ini sangat populer di internet dan kompatibel dengan sebagian besar peramban web terkemuka seperti Google Chrome, Internet Explorer (IE), Mozilla Firefox, Netscape, dan Opera. Kode JavaScript dapat ditambahkan ke halaman web menggunakan tag script. Bersama HTML dan CSS, JavaScript merupakan salah satu teknologi utama yang mendukung World Wide Web. Fungsinya adalah membuat halaman web lebih interaktif dan menjadi elemen penting dalam pengembangan aplikasi web. JavaScript pertama kali dikembangkan oleh Brendan Eich dari Netscape dengan nama awal *Mocha*. Kemudian, namanya berubah menjadi *LiveScript* sebelum akhirnya diberi nama JavaScript. Sebelumnya, Navigator mendukung Java untuk mempermudah penggunaannya oleh pemrogram yang tidak akrab dengan Java. Untuk mendukung kebutuhan ini, Netscape mengembangkan bahasa pemrograman yang diberi nama LiveScript. Bahasa ini terus berkembang hingga akhirnya dinamai JavaScript, meskipun JavaScript tidak memiliki hubungan langsung dengan bahasa Java.

Pada program ini kami menggunakan bahasa pemrograman *javascript* sebagai interaktif untuk memvisualisasikan cara kerja pergerakan *head* pada mesin turing serta visualisasinya juga pada FSA dengan perpindahannya pada tiap state melalui pembacaan inputan.

**BAB II**

**ISI**

**2.1 Studi Kasus 1: Mesin Turing menerima string ganjil dengan inputan {a, b}**

Studi kasus ini melibatkan implementasi Mesin Turing yang menerima string ganjil dengan inputan {a, b}. Mesin Turing digunakan untuk membaca string yang terdiri dari karakter a dan b, dan menentukan apakah jumlah keseluruhan simbol dalam string tersebut ganjil. Mesin ini memiliki empat state (q0, q1, q2, dan q3).

**Deskripsi Studi Kasus**

Studi kasus ini mengilustrasikan implementasi Mesin Turing sederhana untuk memproses string yang terdiri dari simbol {a, b}. Tujuan utama mesin ini adalah menentukan apakah jumlah total simbol dalam string bersifat ganjil.

1. Input: String terdiri dari kombinasi karakter a dan b (contoh: abb, aabbb, dll.).
2. Output: Mesin menerima string jika jumlah simbol dalam string ganjil dan menolaknya jika genap.
3. State: Mesin memiliki empat state:

* q0: State awal (initial state).
* q1: State penerimaan(final state).
* q2: State penerimaan (final state).
* q3: State biasa (bukan final state).

**Definisi Formal**

Berikut merupakan definisi formal dari studi kasus tersebut:

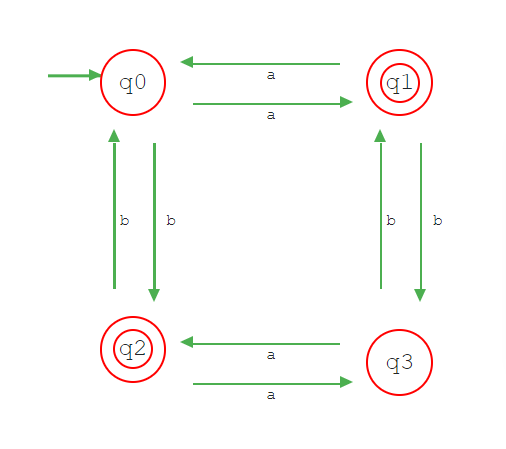
M=(Q,Σ,Γ,δ,S,B​,F​)

Di mana:

* Q = {q0,q1,q2,q3}
* Σ ={a,b}
* Γ = {a,b}
* S = {q0}
* B = [...]
* F = {q1, q2}
* δ (Fungsi Transisi):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State q0 | | | State q1 | | | State q2 | | | State q3 | | | |
| write | move | next | write | move | next | write | move | next | write | move | next |
| a | a | R | q1 | a | R | q0 | a | R | q3 | a | R | q2 |
| b | b | R | q2 | b | R | q3 | b | R | q0 | b | R | q1 |
| [...] | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

Diagram:

****

Mesin akan berhenti ketika tape mencapai **blank** atau disimbolkan “[...]”.

**Desain Aturan Transisi**

**Contoh 1: Mendeteksi string “abbb”.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **b** | **b** | **……..** | **.…….** | **……..** | **……….** |

* Posisi head dimulai pada simbol pertama a dengan state awal q0.
* Mesin membaca simbol a di q0.
* δ(q0,a)=(q1,a,R) dijalankan.
* Mesin tetap menulis a dan head bergerak ke kanan menuju simbol berikutnya.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **b** | **b** | **……..** | **.…….** | **……..** | **……….** |

* Mesin membaca simbol b di q1.
* δ(q1,b)=(q3,b,R) dijalankan.
* Mesin tetap menulis b dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q3​.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **b** | **b** | **……..** | **.…….** | **……..** | **……….** |

* Mesin membaca simbol b di q3.
* δ(q3,b)=(q1,b,R) dijalankan
* Mesin tetap menulis b dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q1​.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **b** | **b** | **……..** | **.…….** | **……..** | **……….** |

* Mesin membaca simbol b di q1​.
* Aturan δ(q1,b)=(q3,b,R) dijalankan.
* Mesin tetap menulis b dan head bergerak ke kanan menuju simbol **blank**.
* Mesin berpindah ke state q3​.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **b** | **b** | **……..** | **.…….** | **……..** | **……….** |

* Mesin membaca simbol **blank** di q3​.
* Tidak ada aturan transisi valid dari q3​ untuk simbol **blank**, sehingga mesin berhenti.
* Mesin berhenti di q3​, yang bukan final state, sehingga string "abbb" tidak diterima.

**Contoh 2: Mendeteksi string “aabbb”.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **a** | **b** | **b** | **b** | **.…….** | **……..** | **……….** |

* Posisi head dimulai pada simbol pertama a dengan state awal q0.
* Mesin membaca simbol a di q0.
* δ(q0,a)=(q1,a,R) dijalankan.
* Mesin tetap menulis a dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q1.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **a** | **b** | **b** | **b** | **.…….** | **……..** | **……….** |

* Mesin membaca a di q1​.
* δ(q1,a)=(q0,a,R) dijalankan.
* Mesin tetap menulis a dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q0​.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **a** | **b** | **b** | **b** | **.…….** | **……..** | **……….** |

* Mesin membaca b di q0​.
* δ(q0,b)=(q2,b,R) dijalankan.
* Mesin tetap menulis b dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q2​.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **a** | **b** | **b** | **b** | **.…….** | **……..** | **……….** |

* Mesin membaca b di q2​.
* δ(q2,b)=(q0,b,R) dijalankan.
* Mesin tetap menulis b dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q0​.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **a** | **b** | **b** | **b** | **.…….** | **……..** | **……….** |

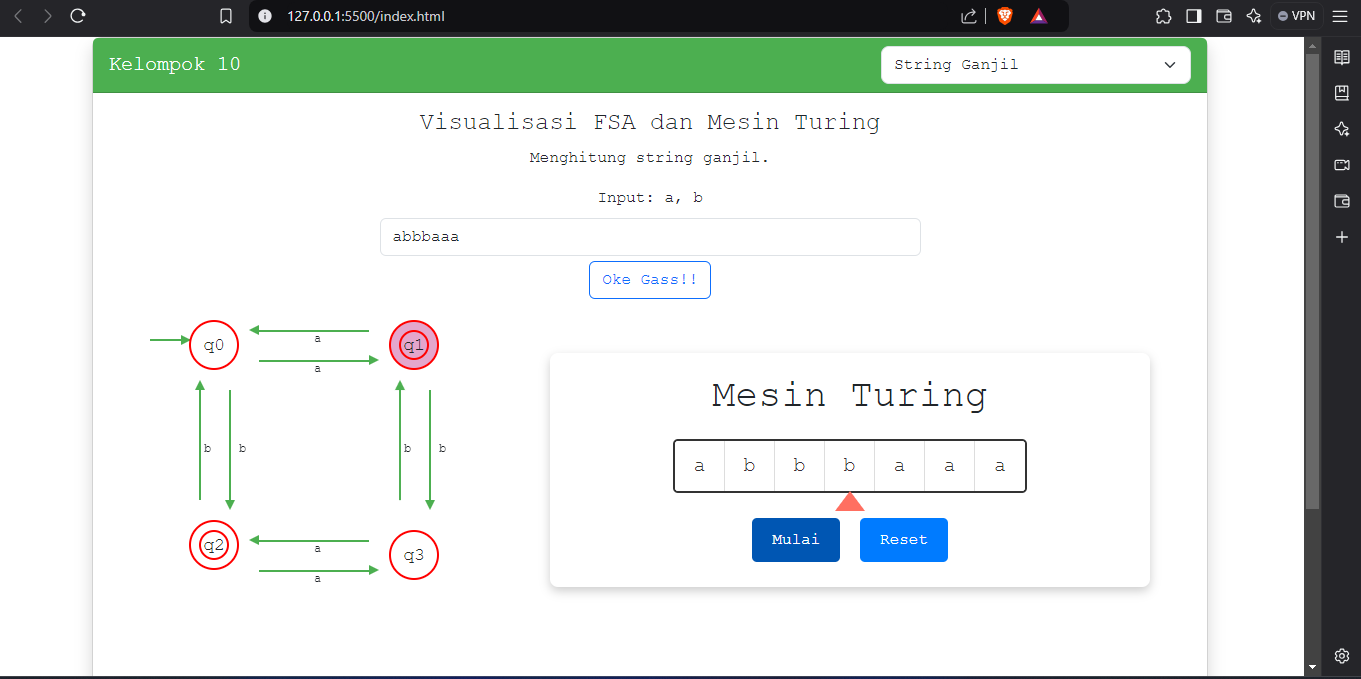
* Mesin membaca b di q0​.
* δ(q0,b)=(q2,b,R) dijalankan.
* Mesin tetap menulis b dan head bergerak ke kanan menuju simbol berikutnya.
* Mesin berpindah ke state q2.

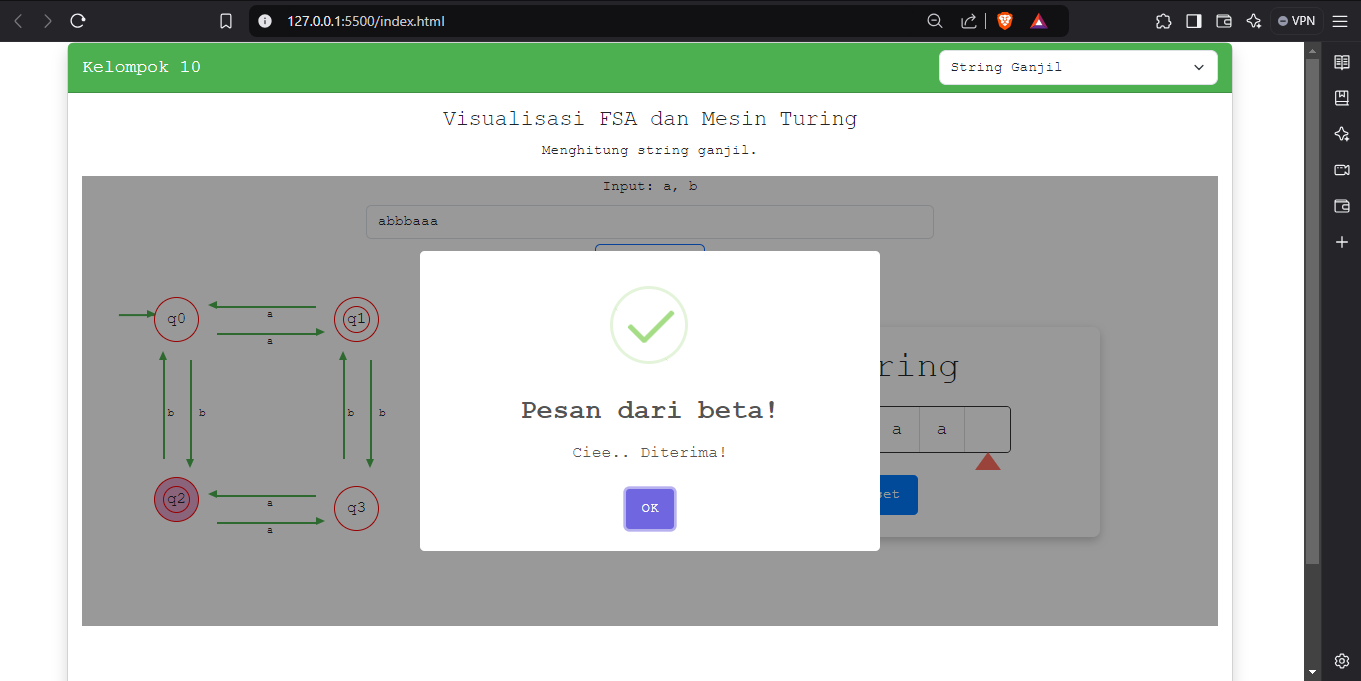
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **a** | **b** | **b** | **b** | **.…….** | **……..** | **……….** |

* Mesin membaca simbol kosong **blank** di q2.
* Tidak ada aturan transisi valid dari q2​ untuk simbol **blank**, sehingga mesin berhenti.
* Mesin berhenti di q2​, yang merupakan final state sehingga string "aabbb diterima".

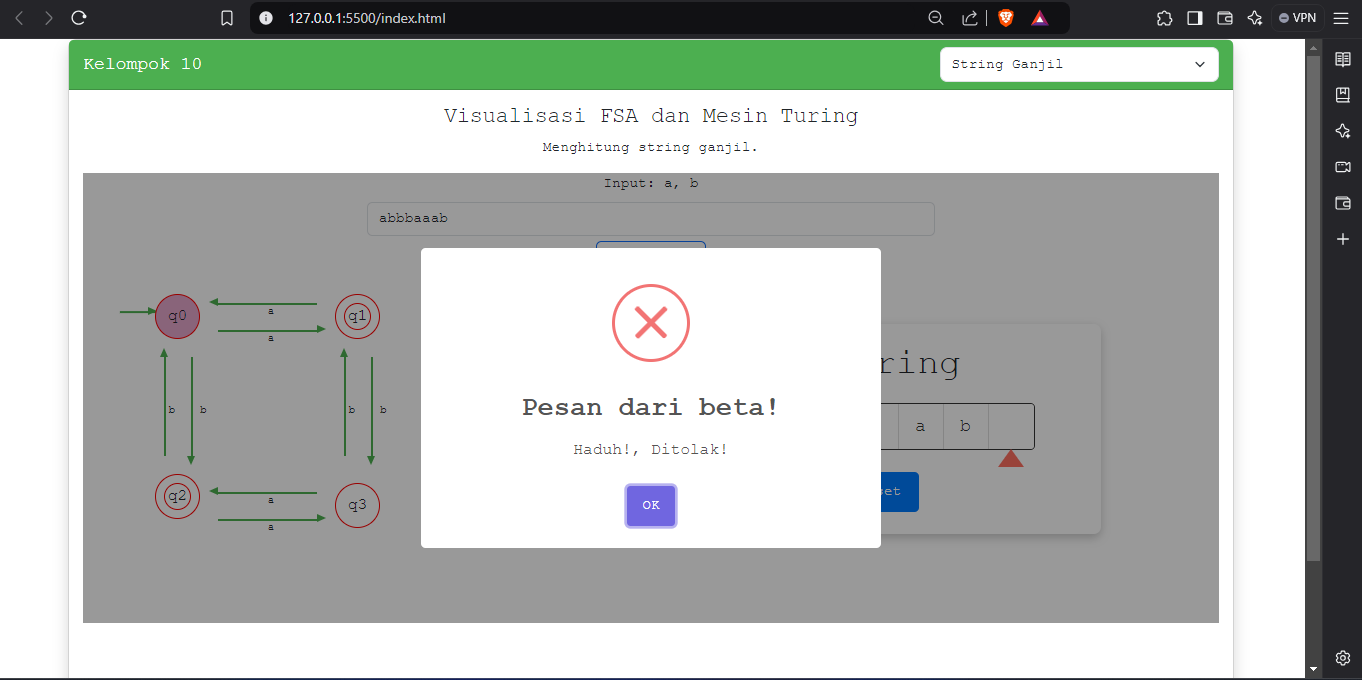
**Antarmuka Program**

Berikut beberapa pengambilan layar pada program.



****

Gambar di atas adalah *alert* ketika mesin turing menerima inputan string ganjil.

****

Gambar di atas adalah *alert* ketika mesin turing tidak menerima inputan string ganjil.

**2.2 Studi Kasus 2: Validasi Email 6 Karakte****r**

**Deskripsi Kasus**

Sebuah perusahaan ingin mengembangkan sistem validasi format email sederhana untuk kebutuhan aplikasi internal. Format email yang diterima sangat spesifik, yaitu hanya terdiri dari 6 karakter dan hanya boleh satu titik dengan format sebagai berikut:

[a-z, 0-9, ., @, g, m, a, i, l, ., c, o, m]

Contohnya:

[a1afyu@gmail.com](mailto:a1afyu@gmail.com)

[alda52@gmail.com](mailto:alda5@gmail.com)

**Aturan format:**

1. Karakter pertama adalah huruf, angka (a-z, 0-9).
2. Karakter kedua adalah huruf, angka, titik, (a-z, 0-9, .).
3. Karakter ketiga adalah huruf, angka, titik, (a-z, 0-9, .).
4. Karakter keempat adalah huruf, angka, titik, (a-z, 0-9, .).
5. Karakter kelima adalah huruf, angka, titik, (a-z, 0-9, .).
6. Karakter keenam adalah huruf, angka (a-z, 0-9).
7. Karakter ketujuh adalah simbol (@).
8. Karakter kedelapan adalah huruf (g).
9. Karakter kesembilan adalah huruf (m).
10. Karakter kesepuluh adalah huruf (a).
11. Karakter kesebelas adalah huruf (i).
12. Karakter kedua belas adalah huruf (l).
13. Karakter ketiga belas adalah titik (.).
14. Karakter keempat belas adalah huruf (c).
15. Karakter kelima belas adalah huruf (o).
16. Karakter keenambelas adalah huruf (m).

**Definisi Formal**

Berikut merupakan definisi formal dari studi kasus tersebut:

M=(Q,Σ,Γ,δ,S,B​,F​)

Di mana:

* Q = {qs, q0, q1, q2 , q3, q4, q5, q6, q7, q8, q9, q10, q11, q12, q13, q14, qf }
* Σ ={a-z, 0-9, .}
* Γ = {a-z, 0-9, .}
* S = {qs}
* B = [...]
* F = {qf}
* δ (Fungsi Transisi) :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State qs | | | State q0 | | | State q1 | | | State q2 | | | |
| write | move | next | write | move | next | write | move | next | write | move | next |
| a-z, 0-9 | sesuai input | R | q0 | sesuai input | R | q1 | sesuai input | R | q2 | Sesuai input | R | q3 |
| . (titik) | Ø | Ø | Ø | . | R | q2 | . | R | q2 | . | R | q3 |
| Sebelum simbol ‘@’ hanya boleh sepanjang 6 karakter | | | | | | | | | | | | |
| @ | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| g | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| a | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| i | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| l | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| . | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| c | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| o | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State q3 | | | State q4 | | | State q5 | | | State q6 | | | |
| write | move | next | write | move | next | write | move | next | write | move | next |
| a-z, 0-9 | sesuai input | R | q4 | sesuai input | R | q5 | Ø | Ø | Ø | Ø | Ø | Ø |
| . (titik) | . | R | q4 | . | R | q5 | Ø | Ø | Ø | Ø | Ø | Ø |
| Sebelum simbol ‘@’ hanya boleh sepanjang 6 karakter | | | | | | | | | | | | |
| @ | Ø | Ø | Ø | Ø | Ø | Ø | @ | R | q6 | Ø | Ø | Ø |
| g | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | g | R | q7 |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| a | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| i | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| l | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| . | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| c | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| o | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

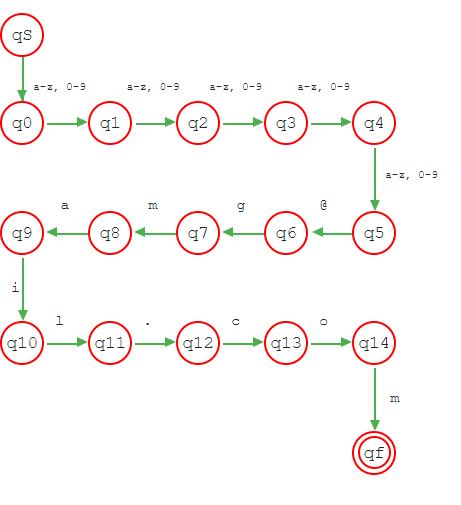
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State q7 | | | State q8 | | | State q9 | | | State q10 | | | |
| write | move | next | write | move | next | write | move | next | write | move | next |
| a-z, 0-9 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| . (titik) | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| Sebelum simbol ‘@’ hanya boleh sepanjang 6 karakter | | | | | | | | | | | | |
| @ | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| g | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| m | m | R | q8 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| a | Ø | Ø | Ø | a | R | q9 | Ø | Ø | Ø | Ø | Ø | Ø |
| i | Ø | Ø | Ø | Ø | Ø | Ø | i | R | q10 | Ø | Ø | Ø |
| l | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | l | R | q11 |
| . | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| c | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| o | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State q11 | | | State q12 | | | State q13 | | | State q14 | | | |
| write | move | next | write | move | next | write | move | next | write | move | next |
| a-z, 0-9 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| . (titik) | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| Sebelum simbol ‘@’ hanya boleh sepanjang 6 karakter | | | | | | | | | | | | |
| @ | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| g | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| a | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| i | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| l | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| . | . | R | 12 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| c | Ø | Ø | Ø | c | R | q13 | Ø | Ø | Ø | Ø | Ø | Ø |
| o | Ø | Ø | Ø | Ø | Ø | Ø | o | R | q14 | Ø | Ø | Ø |
| m | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | m | R | qf |

Keterangan:

Pembacaan simbol “@” hanya akan diterima oleh state q5 dan kemudian terdapat state untuk memvaliditas format sebagai “gmail.com”, ketika state q14 menerima input ‘m’ maka berpindah ke state qf yang di mana tidak akan menerima inputan lagi sehingga proses selesai.

Diagram:



**Desain Aturan Transisi**

**Contoh 1: Mendeteksi string “a1afyu@gmail.com”.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Posisi head dimulai pada simbol pertama a dengan state awal qs.
* Mesin membaca simbol a di qs.
* δ(qs,a)=(q0, a, R) dijalankan.
* Mesin berpindah ke state q0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol 1 di q0
* δ(q0,1)=(q1, 1, R) dijalankan.
* Mesin berpindah ke state q1​.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol a di q1
* δ(q1,1)=(q2, a, R) dijalankan.
* Mesin berpindah ke state q2​.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol f di q2
* δ(q2,f)=(q3, f, R) dijalankan.
* Mesin berpindah ke state q3​

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol y di q3
* δ(q3,y)=(q4, y, R) dijalankan.
* Mesin berpindah ke state q4.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol u di q4
* δ(q4,u)=(q5, u, R) dijalankan.
* Mesin berpindah ke state q5.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol @ di q5
* δ(q5,@)=(q6, @, R) dijalankan.
* Mesin berpindah ke state q6​.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol g di q6
* δ(q6,g)=(q7, g, R) dijalankan.
* Mesin berpindah ke state q7.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m | a |

* Mesin membaca simbol m di q7
* δ(q7,m)=(q8, m, R) dijalankan.
* Mesin berpindah ke state q8

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol a di q8
* δ(q8,a)=(q9, a, R) dijalankan.
* Mesin berpindah ke state q9.



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol i di q9
* δ(q9,i)=(q10, i, R) dijalankan.
* Mesin berpindah ke state q10.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol l di q10
* δ(q10,l)=(q11, 1, R) dijalankan.
* Mesin berpindah ke state q11.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 |  | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol ‘.’ (titik) di q11
* δ(q11,.)=(q12, (titik), R) dijalankan.
* Mesin berpindah ke state q12.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol c di q12
* δ(q12,c)=(q13, c, R) dijalankan.
* Mesin berpindah ke state q13

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol o di q13
* δ(q13,o)=(q14, o, R) dijalankan.
* Mesin berpindah ke state q14.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | u | @ | g | m | a | i | l | . | c | o | m | ... |

* Mesin membaca simbol m di q14
* δ(q14,m)=(qf, m, R) dijalankan.
* Mesin berpindah ke state qf.
* Tidak ada aturan transisi qf, sehingga mesin berhenti.
* Mesin berhenti di qf​, yang merupakan final state sehingga email tersebut dinyatakan valid.

**Contoh 2: Mendeteksi string “a1afy.@gmail.com”.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | . | @ | g | m | a | i | l | . | c | o | m |

* Posisi head dimulai pada simbol pertama a dengan state awal qs.
* Mesin membaca simbol a di qs.
* δ(qs,a)=(q0, a,
* R) dijalankan.
* Mesin berpindah ke state q0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | . | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol 1 di q0
* δ(q0,1)=(q1, a, R) dijalankan.
* Mesin berpindah ke state q1​.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | . | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol a di q1
* δ(q1,a)=(q2, 1, R) dijalankan.
* Mesin berpindah ke state q2​.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | . | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol f di q2
* δ(q2,f)=(q3, f, R) dijalankan.
* Mesin berpindah ke state q3​

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | . | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol y di q3
* δ(q3,y)=(q4, y, R) dijalankan.
* Mesin berpindah ke state q4.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 1 | a | f | y | . | @ | g | m | a | i | l | . | c | o | m |

* Mesin membaca simbol . di q4
* Tidak ada aturan transisi . di q4, sehingga mesin berhenti.
* Mesin berhenti di q4, yang bukan final state, sehingga email tersebut dinyatakan tidak valid.

**2.3 Studi Kasus 3: Mesin Turing melakukan perhitungan biasa dengan inputan {0, 1}**

**Deskripsi Studi Kasus**

Simulasi Mesin Turing untuk menjumlahkan dua bilangan dalam format 000C0000 bertujuan untuk menghitung total jumlah simbol 0 yang terletak di kedua sisi simbol pemisah C pada pita. Dalam simulasi ini, setiap 0 merepresentasikan satu unit nilai, sedangkan C berfungsi sebagai pembatas antara dua bilangan. Mesin Turing akan membaca dan memproses setiap simbol 0, menandainya sebagai X setelah dihitung, lalu secara iteratif memindahkan kepala baca hingga seluruh simbol selesai diproses. Hasil akhir berupa total jumlah 0 dari kedua sisi C, yang ditulis kembali di sisi kiri pita, memberikan representasi visual dari proses penjumlahan. Simulasi ini tidak hanya mengilustrasikan cara kerja Mesin Turing dalam memanipulasi pita, tetapi juga mendemonstrasikan logika transisi antar state untuk menyelesaikan operasi matematika sederhana.

**Definisi Formal**

Berikut merupakan definisi formal dari studi kasus tersebut:

M=(Q,Σ,Γ,δ,S,B​,F​)

Di mana:

* Q = {q0, q1, q2, q3, q4, q5}
* Σ ={0, C, X}
* Γ = {0, C, X, [...]}
* S = {q0}
* B = [...]
* F = {q5}
* δ (Fungsi Transisi):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State q0 | | | State q1 | | | State q2 | | |
| write | move | next | write | move | next | write | move | next |
| 0 | X | R | q1 | 0 | R | q1 | 0 | R | q2 |
| C | [...] | R | q5 | C | R | q2 | Ø | Ø | Ø |
| X | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| [...] | Ø | Ø | Ø | Ø | Ø | Ø | 0 | L | q3 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Simbol Tape | State q3 | | | State q4 | | | State q5 | | |
| write | move | next | write | move | next | write | move | next |
| 0 | 0 | L | q3 | 0 | L | q4 | Ø | Ø | Ø |
| C | C | L | q4 | Ø | Ø | Ø | Ø | Ø | Ø |
| X | Ø | Ø | Ø | X | R | q0 | Ø | Ø | Ø |
| [...] | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

**Contoh: inputan “00C00”.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | C | 0 | 0 | … | … |

**Proses :**

δ (q0, 0) = q1 | write : X | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | … | … |

**Proses :**

δ (q1, 0) = q1 | write : 0 | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | … | … |

**Proses :**

δ (q1, C) = q2 | write : C | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | … | … |

**Proses :**

δ (q2, 0) = q2 | write : 0 | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | … | … |

**Proses :**

δ (q2, 0) = q2 | write : 0 | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | … | … |

**Proses :**

δ (q2, [...]) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | 0 | … |

**Proses :**

δ (q3, 0) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | 0 | … |

**Proses :**

δ (q3, 0) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | 0 | … |

**Proses :**

δ (q3, C) = q4 | write : C | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | 0 | … |

**Proses :**

δ (q4, 0) = q4 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | 0 | … |

**Proses :**

δ (q4, X) = q0 | write : X | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | C | 0 | 0 | 0 | … |

**Proses :**

δ (q0, 0) = q1 | write : X | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | … |

**Proses :**

δ (q1, C) = q2 | write : C | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | … |

**Proses :**

δ (q2, 0) = q2 | write : 0 | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | … |

**Proses :**

δ (q2, 0) = q2 | write : 0 | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | … |

**Proses :**

δ (q2, 0) = q2 | write : 0 | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | … |

**Proses :**

δ (q2, [...]) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | 0 |

**Proses :**

δ (q3, 0) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | 0 |

**Proses :**

δ (q3, 0) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | 0 |

**Proses :**

δ (q3, 0) = q3 | write : 0 | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | 0 |

**Proses :**

δ (q3, C) = q4 | write : C | move : L |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | 0 |

**Proses :**

δ (q4, X) = q0 | write : X | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | C | 0 | 0 | 0 | 0 |

**Proses :**

δ (q0, C) = q5 | write : [...] | move : R |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | X | … | 0 | 0 | 0 | 0 |

**Proses :**

* Tidak ada aturan transisi q5, sehingga mesin berhenti.
* Mesin berhenti di q5​, yang merupakan final state sehingga proses perhitungan selesai.
* Terlihat hasil akhir adalah “0000” yaitu sama dengan 4 bilangan.

**BAB III**

**PEMBAHASAN**

**`**

**Daftar Pustaka**

<https://p2k.stekom.ac.id/ensiklopedia/Mesin_Turing>

<https://id.wikipedia.org/wiki/Mesin_Turing>

<https://ngnextg.wordpress.com/2020/04/23/fsa-dan-nfsa/>

<https://id.wikipedia.org/wiki/JavaScript>

<https://repository.unikom.ac.id/40378/1/Pertemuan4.pptx>

https://www.geeksforgeeks.org/universal-turing-machine/