

Mesin Turing dan *Palindrome*

Fajar Sidik H and 23513186

Program Magister Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

fajar.sidik@students.itb.ac.id

Mesin Turing adalah suatu model yang sangat sederhana dari komputer. Konsep mesin Turing pertama kali di kenalkan oleh Alan Turing pada tahun 1936 dalam papernya yang berjudul "*On Computable Numbers, with an Application to the Entscheidungsproblem*". Konsep mesin Turing inilah yang menjadi dasar dari teori modern komputabilitas. Mesin Turing dapat digunakan untuk menyelesaikan beberapa permasalahan matematis yang sederhana. Contohnya perhitungan pada bilangan bulat, menyalin simbol, menghitung suatu fungsi bilangan bulat dan lain lain. Dalam makalah ini akan di *design* suatu mesin Turing yang akan menerima bahasa *palindrome*. *Palindrome* adalah suatu kata, kalimat yang jika dibaca dari depan ke belakang atau sebaliknya akan sama hasilnya. Akan dibuktikan juga bahwa jika suatu masalah dapat diselesaikan oleh mesin Turing satu pita maka dapat pula diselesaikan dengan mesin Turing banyak pita.

Kata kunci : Alan Turing, Mesin Turing, *Palindrome*, teori komputabilitas

1. Pendahuluan

Pada tahun 1900, D. Hilbert mengeluarkan 23 pertanyaan yang terkenal dalam bidang Matematika. Salah satu yang terkenal adalah *Hilbert's tenth problem* (H10) yaitu menemukan algoritma untuk menyelesaikan masalah *multivariable polynomial* dengan koefisien integer. Untuk menanggapi pertanyaan yang di ajukan oleh Hilbert, maka pada tahun 1936 Alan Turing memperkenalkan definisi pertama pengertian dari algoritma. Dalam makalah yang berjudul : "*On Computable Numbers, with an Application to the Entscheidungsproblem*". Dia merumuskan konsep algoritma dan komputasi dengan mesin atau yang lebih dikenal dengan *Turing Machine*. Dalam konsep ini Turing menggambarkan sebuah mesin yang mampu membaca rangkaian beberapa "nol dan satu" (*binary digit*) yang akan menjelaskan cara penyelesaian masalah matematika, dan menyediakan jawaban yang dibutuhkan. Inti dari mesin ini yang dikemudian hari dikenal sebagai ide tentang sebuah komputer. Mesin ini masih berupa konsep, sampai kemudian diwujudkan dalam bentuk nyata beberapa tahun kemudian. Konsep Mesin Turing inilah yang menjadi dasar dari teori modern komputabilitas. Sehingga Alan Turing dikenal juga secara luas sebagai bapak dari Ilmu Komputer dan Kecerdasan buatan.

Konsep *Turing Machine* ini sebenarnya adalah "*Stored program*" yaitu suatu intruksi dapat diterjemahkan kedalam angka biner yang dapat disimpan, dibaca dan ditulis, dimana merupakan konsep yang sangat penting untuk pembuatan komputer modern saat ini.. Akan tetapi pada tahun 1936 komputer fisik belum ditemukan, sehingga konsepnya hanya sebatas imajinasi matematika saja. Pada tahun yang sama, Turing mendapatkan "*Smith's Prize*" (penghargaan dari *Cambridge University*) untuk pekerjaannya dalam teori probabilitas. Dari September 1936 sampai Juli 1938 ia menghabiskan waktunya belajar dibawah gereja di Universitas *Princeton*. Sebagai tambahan untuk pekerjaan matematikanya, ia mempelajari kriptologi dan membuat tiga dari empat tahap perkalian biner elektro-magnetik. Pada Juni 1938 ia memperoleh gelar PhD dari *Princeton*; Disertasinya, "*Systems of Logic Based on Ordinals*," memperkenalkan konsep Logika *Ordinal* dan *Relative Computing*, dimana mesin Turing diaugmentasikan dengan *Oracle*, sehingga dapat mempelajari soal yang tidak dapat diselesaikan oleh mesin Turing. Kembali ke *Cambridge*, ia mengikuti kuliah oleh *Ludwig Wittgenstein* tentang dasar dasar matematika. Keduanya berdebat dan saling tidak setuju, dimana Turing membela *formalism* sementara Wittgenstein mendorong pandangannya bahwa matematika tidak menemukan kebenaran mutlak tetapi menciptakannya. Ia juga mulai bekerja paruh waktu di *Government Code and Cipher School* (GCCS). Pada saat pecah perang dunia II ia mengambil pekerjaan penuh waktu di kantor pusatnya, *Bletchley Park*. Disana dia memimpin sebuah tim yang merancang sebuah mesin yang dikenal sebagai "*Bombe*" yang berhasil menterjemahkan pesan Jerman. Dia menjadi tokoh terkenal dan eksentrik di *Bletchley*.

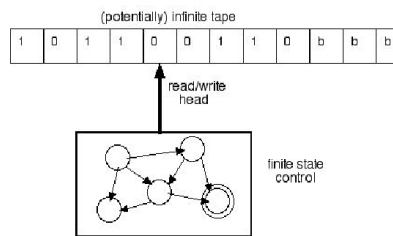
Turing juga mempunyai minat yang sangat besar dalam pengembangan "*Artificial Intelligence*". Untuk itu dia menghabiskan satu tahun di *Cambridge* untuk mempelajari Neurologi dan Fisiologi. Di tahun 1947 dia menulis sebuah paper (tidak pernah diterbitkan selama hidupnya) mengenai konsep yang sekarang dikenal dengan "jaringan neural", dimana serangkaian sistem kompleks mampu memiliki kemampuan belajar. Kemudian tahun 1950 mengeluarkan paper yang berpengaruh besar berjudul "*Computing Machinery and Intelligence*". Dalam papernya ini Turing mengusulkan "Tes Turing" sebagai sebuah metode untuk menentukan apakah sebuah mesin memiliki "*Artificial Intelligence*". Hingga tahun 1990-an Tes ini masih dianggap sebagai cara yang paling baik untuk menentukan intelegensia dari sebuah mesin.

Dibalik segala kesuksesannya, ternyata Turing adalah seorang Homoseksual. Ini sudah dimulai sejak ia masih muda, dan kemudian di tahun 1953 ia ditangkap karena melakukan hubungan seksual dengan seorang pemuda. Dibandingkan masuk penjara, Turing lebih memilih alternatif hukuman suntik Estrogen untuk menetralkan hormonnya. Setelah "kelainannya" ini diketahui oleh publik, Turing kehilangan satu persatu pekerjaannya. Tidak mampu menahan malu dan stress, pada 7 Juni 1954, Turing memilih untuk mengakhiri hidupnya dengan melakukan bunuh diri (memakan apel yang telah dilumuri sianida) di rumahnya, *Wislow, London*.

2. Dasar Teori

2.1 Mesin Turing

Mesin Turing adalah suatu alat komputasi ideal yang memiliki *head* terdiri baca dan tulis (biasa disebut *Finite State Control*) dan sebuah tape/pita yang akan dilaluinya. Pita dibagi menjadi beberapa sel atau kotak yang memiliki panjang tak terhingga .dimana batas kiri tetap dan batas kanan tidak terbatas. Sel pada pita dapat di baca maupun di tulis sedangkan pita yang tidak berisi simbol masukan akan berisi simbol kosong atau *blank* (B). Sebuah mesin Turing adalah merupakan model matematis sederhana untuk komputer. Meskipun sederhana, mesin Turing dapat menggambarkan perilaku komputer *general-purpose*.



Gambar 1. Mesin Turing

Fungsi *state* dari *Finite State Control* pada pita yang akan di lalainya, untuk satu pergerakan akan melakukan :

- Mengganti *state*.bisa berganti ke *state* yang lain atau *state* yang sama
- Menuliskan simbol pada sel pita yang sedang dilalui. Simbol ini akan diganti dengan simbol yang menjadi masukannya
- Menggerakkan *head* ke kiri atau ke kanan

Sebuah mesin Turing dapat di gambarkan sebagai 7 buah tuple dimana $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$:

M = Mesin Turing

Q = Himpunan berhingga *state* dari *finite control state* (q_0, q_1, \dots atau p_0, p_1, \dots)

Σ = Himpunan semua simbol-simbol masukan

Γ = Himpunan simbol-simbol yang muncul di pita, Γ selalu menjadi subset dari Σ

q_0 = *State* awal, anggota dari Q

B = Simbol kosong, anggota dari Γ

F = Himpunan status akhir

δ = Fungsi transisi dari *state*, misalnya $\delta(q, x) = (p, Y, D)$ dimana :

- q adalah *state* dalam sel pita yang dilalui
- x adalah simbol pada pita yang dilalui
- p adalah *state* selanjutnya anggota Q
- Y adalah simbol masukkan yang akan menggantikan nilai dari simbol sel pita
- D adalah arah pergerakan dari *head* ke kiri atau ke kanan (L/R)

2.2 Palindrome

Palindrome adalah suatu kata, kalimat, angka, atau karakter lain yang jika dibaca dari depan dan dari belakang adalah hasilnya sama contoh : *nevereven*, *todderasesareddot*, kasur rusak 1010110101, 9019109 dan sebagainya.

Berikut contoh kata *palindrome* yang dibuat dalam matriks 5x5 dan 6x6 :

| | | | | |
|---|---|---|---|---|
| F | A | R | A | D |
| A | L | E | D | A |
| R | E | B | E | R |
| A | D | E | L | A |
| D | A | R | A | F |

Gambar 2. Palindrome Matriks 5x5

| | | | | | |
|---|---|---|---|---|---|
| E | S | S | S | S | E |
| S | L | E | E | T | S |
| S | E | R | V | E | S |
| S | E | V | R | E | S |
| S | T | E | E | L | S |
| E | S | S | S | S | E |

Gambar 3. Palindrome Matriks 6x6

Sedangkan dalam bilangan untuk membuat suatu *palindrome* adalah sangat sederhana yaitu kita jumlahkan bilangan *palindrome* satu dengan kebalikan *palindrome* lain dan dijumlahkan terus menerus sampai terbentuk bilangan *palindrome*. Misalnya $94 + 49 = 143$, $143 + 341 = 484$

Bilangan *palindrome* juga dapat dibuat dari penjumlahan kuadrat bilangan berurutan misalnya $11^2 + 12^2 + 13^2 = 434$, $11^2 + 12^2 + 13^2 + 14^2 + 15^2 + 16^2 = 1111$, dan sebagainya.

Dalam pemrograman untuk mengetahui apakah suatu bilangan *palindrome* adalah dengan algoritma sederhana sebagai berikut :

- Masukkan bilangan
- Reverse* bilangan tersebut

- c. Bandingkan dengan bilangan yang menjadi masukkannya
- d. Jika sama maka cetak bilangan tersebut
- e. Jika tidak cetak "bukan bilangan *palindrome*"

3. *Design* mesin turing

3.1 *Design* mesin turing satu pita untuk *palindrome* biner
 Pada bagian ini akan di *design* sebuah mesin turing satu pita yang mana akan menerima masukkan bilangan biner *palindrome* 1101011.

Mesin turing pengenalan bahasa $L_{pal} = \{X \in (0,1)^*\}$, dimana L_{pal} adalah *palindrome* dapat ditulis sebagai berikut $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ dimana :

$\Gamma = (0,1,B)$,

$\Sigma = (0,1)$,

$q_0 = i$,

$F = u$,

$B =$ Simbol *blank*

$Q = (i, p_0, p_1, r_0, r_1, t, u)$.

Algoritma yang dibuat adalah :

- a. Inisialisasi yaitu tempatkan *head* pada pita di bagian ujung kiri pada bilangan biner pertama, terima jika pita terdiri dari simbol kosong
- b. Periksa bilangan biner pertama, simpan ganti dengan simbol kosong
- c. Gerakkan *head* ke kanan sampai ketemu simbol kosong periksa bilangan biner yang terakhir jika sama dengan bilangan biner pertama hapus, ganti dengan simbol kosong dan kemudian ke langkah a.
- d. Jika tidak sama berhenti masukan tidak diterima

Fungsi *state*

$i =$ *State* inisial (awal)

$p_0 =$ Mencari simbol 0 pada RS (bagian kanan paling akhir dari *input*)

$p_1 =$ Mencari simbol 1 pada RS

$r_0 =$ Menemukan RS, periksa apakah 0

$r_1 =$ Menemukan RS, periksa apakah 1

$t =$ Kembali ke LS (bagian kiri paling ujung dari *input*)

$u =$ *State* akhir (diterima)

Fungsi transisi yang terjadi

$\delta(i, B) = (u, B, R)$ terima jika pita terdiri dari simbol *blank*

$\delta(i, 0) = (p_0, B, R)$ hapus simbol 0 pada LS, cari 0 pada RS

$\delta(i, 1) = (p_1, B, R)$ hapus simbol 0 pada LS, cari 1 pada RS

$\delta(p_0, 0) = (p_0, 0, R)$

$\delta(p_0, 1) = (p_0, 1, R)$ gerak ke kanan sampai ketemu RS

$\delta(p_1, 0) = (p_1, 0, R)$ gerak ke kanan sampai ketemu RS

$\delta(p_1, 1) = (p_1, 1, R)$

$\delta(p_0, B) = (r_0, B, L)$ ketemu RS

$\delta(p_1, B) = (r_1, B, L)$ ketemu RS

$\delta(r_0, 0) = (t, B, L)$ periksa RS = 0, Hapus simbol 0

$\delta(r_1, 1) = (t, B, L)$ periksa RS = 1, Hapus simbol 1

$\delta(r_0, B) = (u, B, R)$ terima jika pita terdiri dari simbol *blank*

$\delta(r_1, B) = (u, B, R)$ terima jika pita terdiri dari simbol *blank*

$\delta(t, 0) = (t, 0, L)$ gerak ke kiri sampai ketemu LS

$\delta(t, 1) = (t, 1, L)$ gerak ke kiri sampai ketemu LS

$\delta(t, B) = (i, B, R)$ ketemu LS, kembali ke *state* i

| | 0 | 1 | B |
|----------------|-----------------------|-----------------------|-----------------------|
| i | (p ₀ ,B,R) | (p ₁ ,B,R) | (u,B,R) |
| p ₀ | (p ₀ ,0,R) | (p ₀ ,1,R) | (r ₀ ,B,L) |
| p ₁ | (p ₁ ,0,R) | (p ₁ ,1,R) | (r ₁ ,B,L) |
| r ₀ | (t, B,L) | | (u,B,R) |
| r ₁ | | (t, B,L) | (u,B,R) |
| t | (t,0,L) | (t,1,L) | (i,B,R) |

Tabel 1. Fungsi transisi

Rangkaian deskripsi untuk bilangan biner *palindrome* dengan Mesin Turing satu pita 1101011BB...

i 1101011BB | B p₁101011BB | B1p₁01011BB |
 B 10 p₁1011BB | B 101 p₁011BB | B1010 p₁11BB |
 B 10101 p₁1BB | B 101011 p₁BB | B10101r₁1BB |
 B1010t1BBB | B101t01BBB | B10t101BBB |
 B1t0101BBB | Bt10101BBB | tB10101BBB |
 Bi10101BBB | BBp₁0101BBB | BB0p₁101BBB |
 BB01p₁01BBB | BB010 p₁1BBB | BB0101 p₁BBB |
 BB010 r₁1BBB | BB01t0BBBB | BB0t10BBBB |
 BBt010BBBB | BtB010BBBB | BBi010BBBB |
 BBB p₀10BBBB | BBB1p₀0BBBB | BBB10p₀BBBB |
 BBB1r₀0BBBB | BBBt1BBBBB | BBtB1BBBBB |
 BBBi1BBBBB | BBBBp₁BBBBB | BBBr₁BBBBBB |
 BBBBuBBBBB (diterima)

Bagaimanakah membuat mesin turing dengan pita banyak (*Multitape*) yang menerima masukkan *palindrome*?

3.2 *Design* mesin turing tiga pita untuk *palindrome* biner
Design sebuah mesin turing tiga pita yang mana akan menerima masukkan bilangan biner *palindrome* 1101011.

Mesin turing tiga pita pengenalan bahasa *palindrome*:

$L_{pal} = \{X \in (0,1)^* : PAL(X) = 1\}$

3 pita $M = (Q, \Gamma, \delta)$ dimana :

$\Gamma = (0,1, \blacktriangleright, \blacksquare)$,

$\blacktriangleright =$ simbol awal

$\blacksquare =$ simbol *blank*

$Q = (q_{start}, q_{copy}, q_{left}, q_{test}, q_{halt})$,

$X \in (0,1)^*$

$PAL(X) = 1$, jika X adalah *palindrome*

Pita 1 = pita *input*
Pita 2 = pita *work*
Pita 3 = pita *output*
Pergerakan *head* L,R,S = *left*, *right*, *stay*
Dimana pita satu akan menjadi masukkan dari bilangan biner, pita 2 akan memeriksa apakah masukkan adalah *palindrome* sedangkan pita 3 akan memberikan hasil jika bernilai 1 maka masukkan adalah bilangan *palindrome* atau jika bernilai 0 maka masukkan bukan *palindrome*

Algoritma yang dibuat adalah :

- Masukkan bilangan ke pita *input*
- Salin semua bilangan dari pita *input* ke pita *work*
- Gerakkan *head* dari pita *input* ke posisi simbol awal
- Pada setiap gerakan, geser *head* dari pita *input* ke kanan, *head* pada pita *work* ke kiri dan kemudian periksa simbol yang sama

Untuk lebih mudah dipahami akan diberikan penjelasan dengan menggunakan tabel dan gambar

Fungsi transisi δ

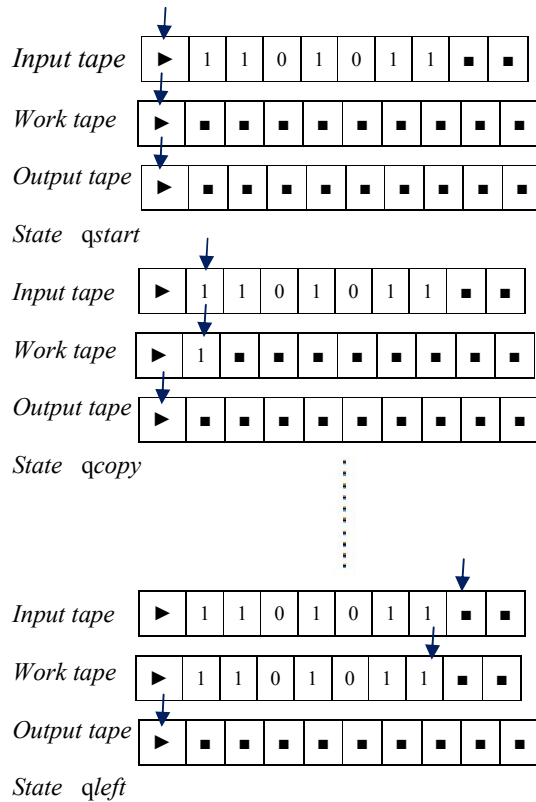
qstart = head pada semua pita pada simbol awal

qcopy = salin nilai dari pita *input* ke pita *work*

| Q | Input tape | work tape | output tape | Q' | (L,R,S) |
|--------|------------|-----------|-------------|--------|---------|
| qstart | ▶ | ▶ | ▶ | qcopy | (R,R,S) |
| qcopy | 1 | 1 | - | qcopy | (R,R,S) |
| qcopy | 0 | 0 | - | qcopy | (R,R,S) |
| qcopy | ■ | - | - | qlleft | (S,S,S) |

Tabel 2. Fungsi transisi qstart dan qcopy

- Inisial awal, posisi tiap *head* pada semua pita adalah simbol awal ▶.
- *Head* dari pita *input* bergerak ke kanan dan berganti *state* menjadi *qcopy*, *head* pada pita *work* gerak ke kanan, *head* pita *output* tidak bergerak tetap pada posisi semula (*stay*)
- Nilai dari pita *input* akan disalin ke pita *work* lalu kedua *head* pita bergerak kekanan sampai *head* pita *input* berada pada simbol ■, pita *output* tetap tidak bergerak
- Jika *head* pada pita *input* ketemu simbol ■ maka *head* pita *work* berhenti dan *state* berganti menjadi *qlleft*, pergerakan *head* tidak ada disetiap pita



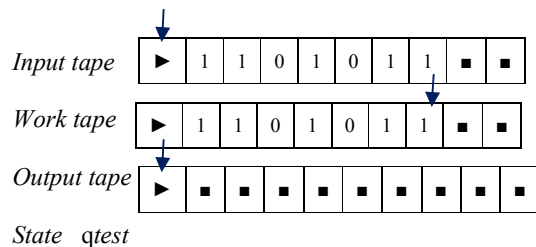
Gambar 4. Pergerakan *head* pada state qstart,qcopy

qlleft = gerakkan *head* pita *input* ke kiri sampai ketemu simbol awal

| Q | Input tape | work tape | output tape | Q' | (L,R,S) |
|--------|------------|-----------|-------------|--------|---------|
| qlleft | 0 | - | - | qlleft | (L,S,S) |
| qlleft | 1 | - | - | qlleft | (L,S,S) |
| qlleft | ▶ | - | - | qtest | (S,S,S) |

Tabel 3. Fungsi transisi qlleft

- *Head* pada pita *input* akan bergerak terus ke arah kiri sampai ketemu simbol ▶. *Head* pita *work* dan pita *output* tetap pada posisi
- Jika *head* pita *input* sudah berada pada simbol ▶ maka *state* berubah menjadi *qtest*



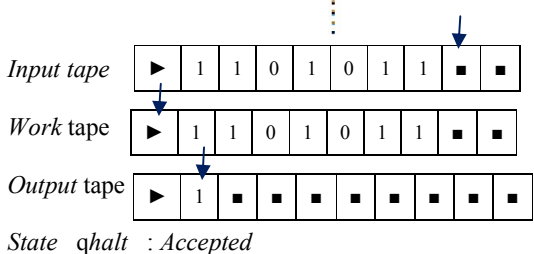
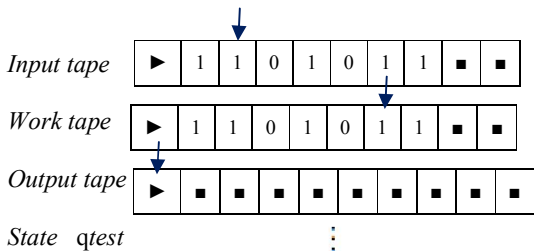
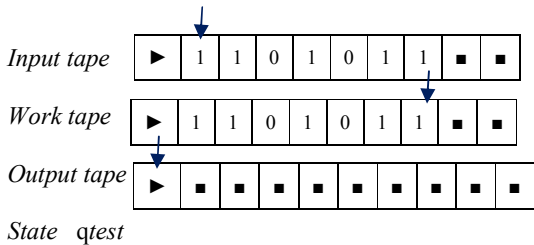
Gambar 5. Pergerakan *head* pada state qlleft

qtest = bandingkan nilai pada pita *input* dengan pita *work*

| Q | Input tape | work tape | output tape | Q' | (L,R,S) |
|-------|------------|-----------|-------------|-------|---------|
| qtest | ▶ | - | - | qtest | (R,S,S) |
| qtest | 1 | 0 | 0 | qhalt | (S,S,S) |
| qtest | 1 | 1 | - | qtest | (R,L,S) |
| qtest | 0 | 1 | 0 | qhalt | (S,S,S) |
| qtest | 0 | 0 | - | qtest | (R,L,S) |
| qtest | ■ | ▶ | 1 | qhalt | (S,S,S) |

Tabel 4. Fungsi transisi qtest

- Gerakkan *head* pita *input* ke kanan kemudian bandingkan nilai dari pita *input* dengan pita *work*
- Jika nilai pita *input* sama dengan pita *work* maka *head* pita *input* akan bergerak terus ke arah kanan dan *head* pada pita *work* akan bergerak ke kiri
- Jika nilai pita *input* tidak sama dengan pita *work* maka berhenti, status berubah ke *state* qhalt dan *head* pada pita *output* bergerak ke kanan dan member nilai 0 yang berarti *input* tidak diterima (bukan *palindrome*)
- Jika *head* pita *input* sudah berada pada simbol ■ dan *head* pita *work* pada simbol ▶, maka *head* pada pita *output* akan bergerak ke kanan dan memberi nilai 1 yang berarti *input* diterima (*palindrome*)



Gambar 6. Pergerakan *head* pada state qtest, qhalt

4. Kesimpulan

Berdasarkan pembahasan dari *design* mesin turing dapat diambil kesimpulan :

- Mesin turing dapat di *design* untuk menyelesaikan permasalahan-permasalahan matematis
- Mesin turing adalah merupakan model matematis sederhana untuk komputer
- Jika suatu permasalahan dapat di modelkan oleh mesin turing satu pita maka dapat pula di buat untuk mesin turing dengan pita banyak (*Multitape*), begitupula sebaliknya.
- Fungsi transisi yang terjadi pada mesin turing banyak pita lebih sedikit daripada dilakukan pada mesin turing satu pita.

REFERENCES

- [1] Copeland, Jack (2000), *Biography of Turing* http://www.alanturing.net/turing_archive/pages/reference%20articles/Bio%20of%20Alan%20Turing.html
Tanggal Akses : 4 Desember 2014, 14.00
- [2] Hodges, Andrew (1995), *Alan Turing: Biography* <http://www.turing.org.uk/bio/part1.html>
Tanggal Akses : 4 Desember 2014, 14.00
- [3] Kowalik, John M (1995), *Alan Turing* <http://ei.cs.vt.edu/~history/Turing.html>
Tanggal Akses : 4 Desember 2014, 14.00
- [4] Spring (2003). *Turing Machine and Reduction* <http://www.cs.toronto.edu/~sacook/csc463h/notes/turing.pdf>
Tanggal Akses : 4 Desember 2014, 14.00
- [5] http://www.fun-with-words.com/palin_example.html
Tanggal Akses : 4 Desember 2014, 20.00
- [6] Munir, Rinaldi (2014), IF 5110 Teori Komputasi – Mesin Turing (Bagian 1) Program Studi Magister Informatika, Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung. 16-17

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 12 Desember 2014

ttd

Fajar Sidik H/ 23513186