

Aplikasi Algoritma *Greedy* Pada Permainan *Big Two*

Ignatius Ronaldo Galman Kurniawan / 13509074

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

ignatius.ronaldo@students.itb.ac.id

Abstrak—*Big Two* atau yang lebih dikenal sebagai *capsa banting* atau *capsa* (berasal dari bahasa *Hokkien*) adalah salah satu permainan kartu yang populer di Indonesia dan dimainkan saat santai bersama teman-teman. Permainan ini menggunakan 52 kartu remi – kartu joker tidak diikutsertakan – dan dibagi masing-masing 13 kartu untuk 4 pemain. Oleh karena itu dinamakan *capsa* yang dalam bahasa *Hokkien* artinya tiga belas. Permainan ini dikenal pula dengan *capsa banting*, karena ada jenis permainan *capsa* lainnya yaitu *capsa susun*. Dalam makalah ini akan dibahas bagaimana permainan *big two* tanpa kita sadari menggunakan salah satu algoritma yang penting dalam bidang teknik informatika, yaitu algoritma *greedy*. Dan algoritma ini cukup penting dalam permainan ini, sehingga dapat dibuat ke dalam program komputer menjadi intelejensia buatan

Kata Kunci—*Big Two*, *capsa banting*, *capsa susun*, *Hokkien*, kartu remi.

1. PENDAHULUAN

Salah satu permainan yang dapat dimainkan secara individu maupun secara kolektif adalah permainan kartu remi. Permainan kartu memiliki banyak sekali jenis dan variasinya. Permainan *Big Two* merupakan salah satu dari jenis permainan kartu remi yang sangat umum dan populer dimainkan di kalangan baik anak-anak, remaja, dan dewasa. Permainan ini biasanya dimainkan untuk mengisi waktu luang untuk membuang kebosanan, permainan ini bisa menggunakan uang (berarti judi), dan tanpa uang (hanya untuk kesenangan). Kemudian permainan ini relatif lebih mudah dimainkan dibandingkan dengan permainan sejenis lainnya, misalkan *poker* ataupun *capsa susun*. Dalam permainan ini dibutuhkan kelicikan, trik, dan ingatan yang baik agar dapat memenangkan pertandingan. Akan tetapi semuanya itu bergantung pada kartu awal yang didapat. Sepintar apapun orang memainkan permainan *Big Two*, jika tidak mendapatkan kartu yang bagus, maka akan hampir mustahil bagi ia untuk memenangkan permainan. Oleh karena banyaknya orang yang memainkan permainan ini, maka permainan ini pun telah hadir dalam perangkat elektronik, dimana kita bisa melawan musuh komputer/*AI* (*Artificial Intelligence*), ataupun melawan musuh dari tempat lain secara *online*. Permainan ini sudah hadir dalam

platform desktop maupun *mobile*. Contoh permainan *desktop* yaitu *viwawa online*, dan dalam *platform mobile* yaitu *Multiplayer Big Two for iPhone/iPad/iPod Touch* dan *Big Big Big 2 for Android*.



Gambar 1 Screenshot Permainan *viwawa online* versi 2.5



Gambar 1 Screenshot Permainan *Multiplayer Big Two*

Karena banyaknya permainan *big two* ini pada berbagai *platform* lah, maka algoritma di balik *AI* nya menjadi menarik untuk dibahas, dan dapat pula diterapkan dalam kehidupan sehari-hari saat bermain *big two* bersama kerabat dan sahabat.

2. DASAR TEORI

2.1 Aturan Permainan *Big Two*

Permainan *capsa* umumnya dimainkan oleh empat orang. Perangkat yang dibutuhkan adalah 52 kartu tanpa *Joker*. Di awal permainan, ke-52 kartu ini dibagikan sampai habis. Jadi setiap pemain mendapat 13 kartu.

Big Two dikenal juga sebagai *capsa* banting, disebut banting karena saat mengeluarkan kartu, sebab saat mengeluarkan kartu, kita membantingnya.

2.1.1 Kartu

Urutan kartu (Terurut menurun)

2-A-K-Q-J-10-9-8-7-6-5-4-3

Keterangan

2-Ace/As-King-Queen-Jack-10-9-8-7-6-5-4-3

Urutan kembang(Terurut menurun)

Spade(sekop) ♠, *Heart*(hati) ♥, *Club*(keriting) ♣, *Diamond* (wajik) ♦ .

2.1.2 Obyektif Permainan

Mengeluarkan seluruh kartu di tangan untuk pertama kali. Pemenang adalah barangsiapa yang mengeluarkan seluruh kartu di tangannya pertama kali.

2.1.3 Kombinasi

Ada 4 tipe/jenis kombinasi yang diizinkan:

- a. *Single/eceran*
Kartu dikeluarkan sendirian. Tanpa kartu lain. Misalnya mengeluarkan 3, atau K, atau apapun.
- b. *Pair/Pasangan*
Kartu dikeluarkan sepasang sekaligus. Syarat mengeluarkan *pair* adalah kedua kartu harus memiliki angka yang sama. Misalnya 3♣ & 3♠. *Pair* yang paling rendah adalah *pair* 3. Sedangkan *pair* yang paling tinggi adalah *pair* 2. Jika *pair*nya sama, maka *pair* yang memiliki kembang paling tinggi yang menang.
- c. *Triple/Threes*
Kartu dikeluarkan tiga sekaligus. Syaratnya adalah ketiga kartu harus memiliki angka yang sama. Misalnya 3♣, 3♥, dan 3♠. *Threes* yang paling rendah adalah *threes* 3, dan *threes* yang paling tinggi adalah *threes* 2.
- d. *Five Card Group/Paket*
Merupakan kombinasi 5 kartu. Berikut kombinasi-kombinasi yang diperbolehkan (diurutkan berdasarkan peringkatnya, dari yang terendah ke tertinggi):
 1. *Straight*/seri

Merupakan kombinasi dengan angka yang berurutan. Misalnya 3-4-5-6-7, atau 9-10-J-Q-K. Untuk menentukan *straight* mana yang lebih tinggi biasanya yang dipilih adalah kartu terakhir di dalam *straight* tersebut. Misalnya 3-4-5-6-7 lebih rendah daripada 9-10-J-Q-K. Karena kartu terakhir dari *straight* 3-4-5-6-7 adalah 7, sedangkan kartu terakhir *straight* 9-10-J-Q-K adalah K. Tapi penentuan berdasarkan kartu terakhir tidak mutlak. Ada banyak variasi. Baca bagian variasi.

2. *Flush*

Merupakan kombinasi dengan kembang yang sama. Nama *flush* ditentukan dari kembangnya. *Flush* yang memiliki angka tertinggi menang dari lawannya. Jika angka tertingginya sama, baru dilihat dari kembangnya. Misalnya 3-4-8-J-A dengan kembang sekop lebih tinggi dari 5-9-10-Q-K dengan kembang hati.

3. *FullHouse/Polo*

Merupakan kombinasi antara *pair* dan *threes*. Nama polo ditentukan dari *threes*nya. Misalnya *polo Queen* adalah 3-3-Q-Q-Q. Untuk menentukan *polo* mana yang lebih tinggi dilihat dari *threes*nya. Misalnya kombinasi *threes Ace & pair 3* akan menang melawan *threes King & pair Jack*. Walaupun *pair Jack* lebih tinggi dari *pair 3*.

4. *Four of a kind/Piting/Siki/Quad*

Merupakan kombinasi empat kartu berangka sama dengan 1 kartu acak (bebas). Nama *piting* ditentukan dari empat kartu yang berangka sama. Misalnya *piting Queen* adalah Q-Q-Q-Q-3. Ketika kombinasi *piting* dikeluarkan, 2 akan dibom. Maksudnya adalah dalam permainan satuan, 2 akan menjadi yang terendah, sedangkan *Ace* tertinggi. Untuk mengembalikan 2 menjadi satuan tertinggi, diperlukan *piting* lagi untuk kembali mengebom.

Jika pada awal permainan pemain mendapatkan *siki* 2, maka ia dinyatakan sebagai pemenang.

Rasio kemungkinan untuk mendapatkan *siki* 2 adalah

$$\frac{\binom{48}{9}}{\binom{52}{13}} = 0.264106\% = 1 : 379$$

5. *Straight Flush*/seri kembang
Merupakan kombinasi 5 kartu berurutan. Tapi kembangnya juga sama. Misalnya 5-6-7-8-9. Untuk menentukan mana seri kembang yang lebih tinggi lihat dulu angkanya. Baru kemudian lihat kembangnya. Seri kembang tertinggi adalah 10-J-Q-K-A.

6. *Royal Straight Flush*
Sama dengan *Straight Flush*, tapi yang membedakan susunannya adalah 10-J-Q-K-A dengan kembang sekop.

7. *Capsa/Dragon/Pure Dragon*
Pemain mendapatkan 3-4-5-6-7-8-9-10-J-Q-K-A-2 saat pertama kali kartu dibagi, dan pemain tersebut langsung dinyatakan sebagai pemenang, kembang boleh tidak sama.

Rasio untuk mendapatkan *Dragon*:

$$\frac{\binom{4}{1}^{13}}{\binom{52}{13}} = 0.010568\% = 1 : 9462$$

8. *Suited Dragon*
Sama dengan *Dragon*, akan tetapi ketigabelas kartunya memiliki kembang yang sama.

Rasio untuk mendapatkan *Suited Dragon*:

$$\frac{4}{\binom{52}{13}} = 0.00000000629908\% = 1 : 15875338900$$

2.1.4 Cara Bermain

Di awal permainan, setiap pemain mendapatkan 13 kartu. Pemain yang berhak untuk jalan pertama kali adalah pemain yang memiliki kartu terkecil yaitu 3♦. Kartu 3♦ ini harus dikeluarkan di awal permainan, boleh satuan ataupun kombinasi. Permainan berlanjut searah jarum jam/berlawanan arah jarum jam.

Bila salah seorang pemain mengeluarkan kartu satuan, maka pemain berikutnya juga harus melawannya dengan satuan yang lebih tinggi. Misalnya 3 dilawan dengan A. Satuan harus dilawan satuan. *Pair* harus dilawan *pair*. *Threes* harus dilawan *threes*. Dan paket harus dilawan paket. Bila pemain tersebut tidak punya kartu yang bisa melawannya, atau tidak ingin mengeluarkan kartu bisa melakukan *pass* atau lewat. Setelah *pass*, pemain tersebut tidak boleh mengeluarkan kartu lagi sampai trik baru dimulai (mulai lagi dari awal).

Ketika salah seorang pemain berhasil terus mengeluarkan kartu sampai tidak ada pemain lagi yang bisa melawan, dia

berhak memulai trik baru. Trik baru tidak harus sama dengan trik sebelumnya. Jadi misalnya sebelumnya mengeluarkan satuan, di trik baru bisa mengeluarkan paket atau *pair*.

Permainan umumnya selesai jika salah seorang pemain sudah menghabiskan kartunya. Tetapi terkadang, permainan terus dilanjutkan sampai tinggal satu pemain yang masih memiliki kartu. Ketika seorang pemain sudah mengeluarkan kartu terakhirnya, misalnya Q♥. Maka pemain di sebelahnya harus melanjutkan trik tersebut. Jika tidak bisa melanjutkan trik tersebut, maka pemain sesudahnya lagi yang akan melanjutkan triknya. Jika tidak ada pemain yang bisa melawan kartu terakhir tersebut, maka pemain sesudah pemain yang mengeluarkan kartu terakhir mendapat hak warisan. Jadi pemain tersebut berhak memulai trik baru.

2.1.5 Variasi

Permainan *Big Two* ini adalah permainan yang telah memiliki sangat banyak variasi dan versi dalam praktik permainannya, sehingga sebelum memainkan permainan ini hendaknya dibuat suatu perjanjian/konvensi dahulu, karena aturan internasional yang benar-benar baku tidak ada.

Berikut disajikan beberapa contoh variasi permainan *Big Two* yang cukup umum ditemui:

1. Kartu 2 adalah kartu tertinggi dalam permainan satuan. Tetapi bila dikombinasikan dalam *pair*, *threes*, ataupun paket, kartu 2 akan menjadi kartu yang terendah. Jadi misalnya *threes* 3 mengalahkan *threes* 2.
2. Seri tidak harus 5 kartu. Seri 3 kartu, seri 4 kartu, dan seri berapa kartu pun diperbolehkan kecuali 2 kartu. Pemain berikutnya juga harus melawan dengan jumlah kartu yang sama. Misalnya seri 6 kartu dilawan dengan seri 6 kartu.
3. Karena kartu 2 dianggap kartu tertinggi, maka seri 2-3-4-5-6 tidak diperbolehkan.
4. Menentukan mana kartu seri yang paling tinggi ditentukan dari kartu tertingginya. Misalnya: 2♥-3-4-5-6 mengalahkan A-2♦-3-4-5. Ada juga yang menentukan mana kartu seri yang paling tinggi dari kartu terdepennya.
5. Menentukan mana *flush* yang lebih tinggi bukan dilihat dari kartu tertingginya, melainkan kembangnya. Misalnya: *flush* sekop dengan kartu tertinggi K mengalahkan *flush* hati dengan kartu tertinggi Ace.
6. *Piting* dapat dikeluarkan tanpa satu kartu acak.
7. Pemain yang mendapatkan 6 *pair* ditambah 1 kartu acak dapat dinyatakan langsung menang pada awal permainan (sama seperti *capsa*).
8. Pemain yang sudah *pass* masih diperbolehkan mengeluarkan kartu pada trik yang sama.
9. Pemain yang jalan pertama adalah pemain yang menang di babak sebelumnya, bukan yang memiliki 3♦.

10. Setelah satu trik berakhir, seluruh kartu tersebut ditutup agar pemain tidak bisa menghitung kartu. Menghitung kartu maksudnya di sini adalah menghitung kartu setiap lawan dan mengingat kartu apa saja yang sudah keluar. Dengan menghitung jumlah kartu lawan dan mengingat kartu apa saja yang sudah keluar, kita dapat mencari strategi yang baik sehingga peluang menang lebih besar. Misalnya, kartu lawan tinggal empat. Maka kita tidak perlu ragu-ragu paket kita akan dilawan oleh pemain lain.

2.2 Algoritma Greedy

2.2.1 Tentang Algoritma Greedy

Algoritma *greedy* adalah algoritma yang biasanya digunakan untuk mencari solusi terbaik (optimum : maksimum atau minimum) dari sekumpulan kemungkinan/alternatif solusi yang mungkin dan algoritma yang paling populer saat ini.

Secara harafiah, *greedy* artinya rakus atau tamak. Sesuai dengan artinya, prinsip *greedy* adalah dapatkan yang bisa kamu dapat sekarang (*take what you can get now*).

Algoritma *greedy* membentuk solusi langkah per langkah (*step by step*). Terdapat banyak pilihan yang perlu di eksplorasi pada setiap langkah solusi, karenanya pada setiap langkah harus dibuat keputusan yang terbaik dalam menentukan pilihan. Keputusan yang telah diambil pada suatu langkah tidak dapat diubah lagi pada langkah selanjutnya. Sebagai contoh, jika kita menggunakan algoritma *greedy* untuk menempatkan komponen diatas papan sirkuit, sekali komponen telah diletakkan dan dipasang maka tidak dapat dipindahkan lagi.

Pada setiap langkah diperoleh optimum lokal. Bila algoritma berakhir, kita berharap optimum lokal menjadi optimum global.

2.2.2 Kerangka Algoritma Greedy

Skema Umum Algoritma *Greedy*

```
while (problem belum selesai)
{
  buat pilihan terbaik yang feasible
}
```

Algoritma *greedy* disusun oleh elemen-elemen berikut:

- Himpunan kandidat.
Berisi elemen-elemen pembentuk solusi.
- Himpunan solusi
Berisi kandidat-kandidat yang terpilih sebagai solusi persoalan.

- Fungsi seleksi (*selection function*)

Memilih kandidat yang paling memungkinkan mencapai solusi optimal. Kandidat yang sudah dipilih pada suatu langkah tidak pernah dipertimbangkan lagi pada langkah selanjutnya.

- Fungsi kelayakan (*feasible*)

Memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak, yakni kandidat tersebut bersama-sama dengan himpunan solusi yang sudah terbentuk tidak melanggar kendala (*constraints*) yang ada. Kandidat yang layak dimasukkan ke dalam himpunan solusi, sedangkan kandidat yang tidak layak dibuang dan tidak pernah dipertimbangkan lagi.

- Fungsi obyektif, yaitu fungsi yang memaksimalkan atau meminimumkan nilai solusi (misalnya panjang lintasan, keuntungan, dan lain-lain).

3. ANALISIS

Ide dari memainkan permainan *big two* adalah menjadi orang pertama yang mengeluarkan semua kartu di tangannya. Prinsip ini dapat menggunakan algoritma *greedy*, dan *greedy* akan dinamai *greedy by* mengeluarkan kartu paling banyak. Ide dari algoritma ini adalah mengeluarkan kartu sebanyak mungkin dalam giliran/*turn* kita dan sebisa mungkin menghindari *pass*, karena *pass* sangat merugikan kita, sebab jika kita *pass*, maka kita tidak mengeluarkan kartu dan musuh mengeluarkan kartu, kemungkinan kita memenagkan pertandingan akan semakin kecil.

Skema (gambaran garis besarnya) dari algoritma *greedy by* mengeluarkan kartu terbanyak adalah sebagai berikut:

```
function BigTwo(input k:himpunankartu)
→ himpunankartu
{Kamus}
x : kartu
s : himpunankartu
{Algoritma}
while not Solusi(s) and c ≠ 0 do
  x ← Seleksi(c)
  c ← c - {x}
  if Layak(s or {x}) then
    s ← s or {x}
  endif
endwhile
{Solusi(s) atau c = 0}

if Solusi(s) then
  → keluarkan kartu
else
  → pass
```

Penjelasan ide Algoritma :

Fungsi BigTwo akan menerima sebuah *input* yaitu sekumpulan kartu (dalam hal ini kartu di tangan).

Pertama-tama BigTwo akan melakukan *looping*/pengulangan terhadap himpunan kartu yang masih ada dengan berusaha mencari kemungkinan kartu terbanyak dahulu yang akan dikeluarkan dengan fungsi Seleksi, kemudian himpunan kandidat akan dikurangi sebanyak 1 buah.

Fungsi Seleksi ini memiliki peran yang sangat penting dan salah satu dari faktor utama dalam menentukan kesuksesan permainan. Karena jika pilihan kartu yang akan keluar menyebabkan kerugian untuk pemain, maka kemungkinan memenangkan permainan akan semakin kecil. Fungsi seleksi memiliki ide algoritma sebagai berikut:

Dari sekumpulan kartu yang ada, akan diperiksa apakah:

```
if ada paket then
  if ada lebih dari satu paket then
    → paket dengan nilai terendah
  else
    → paket yang ada
else if ada threes then
  if ada lebih dari satu threes then
    → threes dengan nilai terendah
  else
    → threes yang ada
else if ada pair then
  if ada lebih dari satu pair then
    → pair dengan nilai terendah
  else
    → pair yang ada
else if ada eceran
  if ada lebih dari satu eceran
    → eceran dengan nilai terendah
  else
    → eceran yang ada
else
  → not solusi
```

Artinya fungsi seleksi akan memilih dengan urutan prioritas pilihan:

- Paket
- *Threes*
- *Pair*
- Satuan
- *not* solusi / *Pass*

Dan apabila dari urutan tersebut ada kemungkinan lebih dari satu (kecuali *pass*), maka akan dipilih yang lebih rendah dengan harapan apabila musuh bisa melawan kartu kita, kita masih memiliki kartu yang bagus di tangan.

Dan khusus untuk paket, karena ada banyak jenisnya, maka prioritas untuk pemilihan paket pun juga akan mulai dari yang terendah.

Prioritas Paket:

- *Straight*
- *Flush*
- *Full House*
- *Four of a Kind*
- *Straight Flush*
- *Royal Staright Flush*

Pass hanya dipilih apabila tidak ada lagi kartu yang layak untuk dikeluarkan.

Dan setelah suatu kombinasi kartu dipilih, maka itu disebut solusi sementara, solusi sementara ini akan diperiksa oleh fungsi Layak

Solusi sementara yang ada akan diolah oleh fungsi Layak apakah sudah memenuhi ataukah belum untuk dijadikan solusi final.

Layak dalam hal ini memiliki 2 kondisi:

1. Apakah jalan pertama/memulai trik baru:
Jika kita jalan pertama, maka akan dicek apakah di dalam kartu terdapat 3♦, jika tidak ada, maka solusi sementara ditolak.
Jika kita memulai trik baru, maka kartu apapun yang dihasilkan oleh fungsi seleksi akan layak untuk dikeluarkan.
2. Apakah dalam posisi untuk melawan:
Jika kita dalam posisi melawan, maka pertama-tama akan dicek, apakah solusi sementara memiliki tipe yang sama dengan kartu yang dikeluarkan oleh lawan sebelumnya (apakah paket, *threes*, *pair*, ataukah eceran). Jika tipenya sama, baru dicek apakah nilainya sudah lebih tinggi dari kartu lawan sebelumnya, jika iya, maka fungsi layak akan mengembalikan *true* dan solusi sementara pun berubah menjadi solusi.

Skema algoritmanya adalah sebagai berikut:

```
if jalan pertama then
  if ada 3♦ then
    → true
  else
    → false
else if mulai trik baru then
  → true
else if melawan then
  if tipe sama then
    if nilai lebih tinggi then
      → true
    else
      → false
  else
    → false
```

Setelah solusi berhasil didapatkan, maka akan dicek apakah solusi tersebut sudah merupakan solusi atau bukan, jika iya, maka kartu akan dikeluarkan/lawan, jika bukan solusi, maka pemain akan melakukan pass/lewat.

Demikianlah algoritma *greedy* berproses untuk satu giliran/*turn*. Dan untuk giliran-giliran berikutnya algoritma ini akan dipanggil lagi dan mengalami proses yang sama.

Elemen-elemen Algoritma *Greedy* dalam permainan ini adalah:

- Himpunan Kandidat
Semua kartu yang masih di tangan.
- Himpunan Solusi
Kartu-kartu yang mungkin keluar, mungkin paket, *threes*, *pair*, *eceran*, atau *pass*.
- Fungsi Seleksi
Pilih kartu terbanyak yang mungkin keluar dalam satu giliran/*turn*.
- Fungsi Kelayakan
Langkah yang layak adalah salah satu dari paket, *threes*, *pair*, *eceran*, atau *pass*.
- Fungsi Obyektif
Mengeluarkan kartu sebanyak mungkin dalam satu giliran/*turn*.

4. KESIMPULAN

Big Two memang permainan kartu yang asyik untuk dimainkan, karena tidak terlalu sulit dan diperlukan taktik yang jitu untuk memenangkan permainan.

Apalagi di tengah zaman yang serba elektronik ini, permainan ini pun sudah diadopsi ke perangkat yang mampu melakukan proses komputasi, sehingga sebuah algoritma untuk menciptakan *AI* dari pemain permainan ini pun dibutuhkan. Salah satu algoritma yang cukup mangkus dipakai dalam permainan ini adalah algoritma *greedy*. Algoritma *greedy* akan melakukan proses dan waktu eksekusi yang jauh lebih sedikit dibandingkan algoritma *brute force* yang mencoba seluruh kemungkinan dari pengeluaran kartu.

Akan tetapi pula patut diperhatikan bahwa sebegas apapun algoritmanya, jika pemain mendapatkan kartu yang tidak bagus, maka akan sulit untuk bisa memenangkan permainan.

Penulis merekomendasikan penggunaan algoritma ini untuk direalisasikan baik dalam kehidupan sehari-hari maupun dalam program agar dapat memperoleh kemenangan dalam permainan ini.

5. REFERENSI

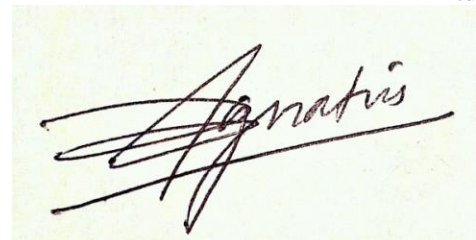
- [1] <http://www.kaskus.us/showthread.php?p=95122307>
- [2] <http://www.viwawa.com/forums/indonesia.8862.html>
- [3] <http://toga161.blogspot.com/2011/01/capsa-banting-dan-aturannya.html>
- [4] http://en.wikipedia.org/wiki/Big_Two
- [5] <http://www.pagat.com/climbing/bigtwo.html>
- [6] <http://herucr.com/algoritma-greedy.html>
- [7] <http://zafnatpaneyah.blogspot.com/2010/07/algoritma-greedy.html>
- [8] http://en.wikipedia.org/wiki/Greedy_algorithm

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, 07 Desember 2011

Ttd



Ignatius Ronaldo Galman Kurniawan / 13509074