

# Aplikasi Algoritma Greedy pada Permainan Kartu Truf Pass

Dimpos A.G. Sitorus / 13513083  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
dimpossitorus@gmail.com

**Abstract**—Makalah ini berisikan penerapan salah satu algoritma yang diajarkan pada mata kuliah IF2211 Strategi Algoritma, yaitu algoritma Greedy pada permainan kartu truf pass

**Index Terms**—permainan kartu truf pass, truf pass, greedy, strategi algoritma, strategi permainan kartu

## I. PENDAHULUAN

### A. Latar Belakang

Permainan kartu adalah salah satu permainan yang cukup banyak peminatnya di dunia. Cukup banyak strategi yang digunakan untuk memenangkan permainan kartu. Salah satunya adalah permainan truf pass. Penulis tertarik menyelesaikan permainan kartu truf pass ini dengan menggunakan algoritma greedy, dimana tujuan utama permainan kartu ini adalah menarik sebanyak mungkin kartu - paling sedikit sesuai dengan penawaran - untuk bisa memenangkan permainan ini.

### B. Tujuan

Tujuan penulis membuat makalah tentang topik ini adalah:

- Menambah pemahaman tentang algoritma Greedy
- Mengimplementasikan algoritma greedy
- Menambah pemahaman tentang permainan kartu strategi, terutama truf pass
- Langkah awal pengembangan kecerdasan buatan untuk pengembangan permainan truf pass pada platform.

## II. TRUF PASS

### A. Deskripsi Singkat Truf Pass

Truf pass adalah sebuah permainan kartu strategi yang cukup umum di Indonesia. Permainan ini bisa dikatakan adalah modifikasi dari permainan truf.

Jika pada permainan kartu truf, pemain bermain sendiri untuk memenangkan permainan, maka pada truf pass, pemain bekerja sama, dua vs dua. Pemain yang satu tim adalah pemain yang bersebarangan secara diagonal. Setiap orang mendapat 13 kartu acak dari satu set kartu remi tanpa kartu joker. Dan seperti truf, pemain hanya mengetahui susunan kartu milik sendiri, bahkan kartu rekanan juga tidak kita ketahui.

Pada awalnya permainan dimulai dengan melakukan penawaran. Untuk permainan pertama biasanya yang mengocok kartu yang memulai penawaran, untuk permainan selanjutnya, yang memenangkan permainan sebelumnya yang memulai penawaran.

Pemenang permainan adalah tim yang berhasil menarik kartu sejumlah lebih besar atau sama dengan penawaran yang dilakukannya.

### B. Urutan Kekuatan Kartu

Urutan kekuatan kartu sama dengan urutan pada truf, yaitu untuk bunga dimulai dari keriting (♣), diamond (♦), heart (♥), dan spade (♠). Untuk urutan angka dimulai dari 2,3,4,5,6,7,8,9,10, Jack, Queen, King, As(Ace).

Kartu 7♠ lebih kuat daripada 7♥, 7♥ lebih kuat daripada 7♦. 7♦ lebih kuat daripada 7♣. Kartu 8♣ lebih kuat daripada 7♠, 9♣ lebih kuat daripada 8♠ dan seterusnya.

### C. Penawaran

Jenis-jenis penawaran ada 5 yaitu, n keriting, n diamond, n heart, n spade dan n no truf, dimana n adalah jumlah yang akan ditarik dan no truf adalah jenis bunga tertinggi. Penawaran selalu dimulai dengan nilai 7 dan bunga yang diinginkan. Penawaran ini berfungsi sebagai penentu permainan. Angka yang disebut akan menjadi jumlah kartu yang akan ditarik, dan bunga yang disebut akan menjadi kartu truf pada permainan. Pemenang penawaran ini

adalah tim/pasangan yang menawarkan yang tertinggi (angka dan bunga tertinggi dalam penawaran). No truf sendiri adalah penawaran dimana permainan dimainkan tanpa kartu truf. Untuk pasangan pemenang penawaran, jumlah kartu yang harus ditarik adalah sejumlah  $n$  kartu, dan untuk pasangan lain adalah sejumlah  $(14 - n)$  kartu. Jika misalnya penawaran tertinggi adalah 8 heart, maka permainan dimulai dengan kartu truf pada heart, jumlah kartu yang harus ditarik oleh pasangan pemenang penawaran adalah 8 kartu, dan jumlah kartu yang ditarik tim yang kalah penawaran adalah 6 kartu. Dan juga ada sistem double dan redouble. Double disebutkan oleh lawan. Redouble disebutkan oleh rekanan, jika tim lawan menyebutkan double. Double disini maksudnya mengalikan skor yang didapat oleh tim yang akan memenangkan permainan dengan dua. Redouble maksudnya mengalikan skor hasil double dengan dua, atau sama dengan mengalikan skor tim yang akan memenangkan pertandingan dengan 4.

#### D. Aturan Bermain, Konvensi dan Scoring

Permainan ini dimulai dengan pemain pertama menurunkan kartu. Boleh dimulai dari kartu yang bukan truf, dan juga boleh dimulai dengan kartu truf. Jika yang diturunkan bukan kartu truf, maka kita harus menurunkan kartu yang bunganya sesuai. Jika kita tidak mempunyai kartu pada bunga tersebut (sering disebut void), maka kita boleh menurunkan kartu truf, atau kita juga boleh 'menyampah' atau menurunkan kartu lain yang bukan truf. Biasanya ini tergantung strategi dan kondisi permainan. Jika pemain pertama menurunkan kartu truf, kita harus menurunkan kartu truf, jika tidak ada, maka kita melakukan tindakan 'menyampah'. Jika kita menurunkan kartu truf maka kita harus menurunkan dengan posisi terbalik, tidak menunjukkan angkanya.

Jika seorang pemain mendapat kartu hanya angka saja, tidak terdapat gambar (Jack, Queen, King, atau As) maka terjadi void dan dilakukan pengocokan ulang. Namun jika void yang terjadi adalah salah satu jenis bunga kartu tidak ada, namun pada bunga lain terdapat kartu gambar (Jack, Queen, King, atau As), permainan tetap dilanjutkan.

Ada beberapa konvensi umum/dasar yang berlaku untuk permainan truf ini. Pada umumnya pemain pertama memainkan kartu kecil dengan bunga yang ditawarkan oleh rekanannya. Misalnya rekanannya menawar kartu keriting, maka pemain pertama menurunkan kartu terkecil yang dimilikinya pada bunga keriting. Untuk orang kedua biasanya mengeluarkan kartu terkecil yang dimilikinya pada bunga kartu yang sedang dimainkan, namun tidak

menutup kemungkinan mengeluarkan kartu tertinggi jika kartu tersebut memang kartu tertinggi ketika permainan sedang berlangsung, contoh As; King, jika As sudah turun; Queen jika King dan As sudah turun. Untuk pemain ketiga pada umumnya melakukan 'jihad', atau menurunkan kartu tertinggi yang dimilikinya. Namun tidak menutup kemungkinan menurunkan kartu rendah/terendah jika ternyata pemain kedua telah menurunkan kartu tertinggi. Pemain keempat biasanya bebas menurunkan kartu apa saja. Jika pemain ketiga menurunkan kartu tertinggi dan pemain keempat masih memiliki kartu yang lebih tinggi maka kartu tersebut akan diturunkan. Namun jika tidak ada kartu yang lebih tinggi, atau pemain kedua (rekanannya) telah menurunkan kartu terendah maka biasanya diturunkan kartu terendah.

Untuk sistem penilaian, ada nilai kecil dan besar. Nilai kecil didapat dari jumlah kartu yang ditarik oleh tim yang memenangkan permainan. Misalnya dimenangkan oleh tim A dengan jumlah kartu yang ditarik adalah 8, maka skor disebut 8-0. Pada permainan selanjutnya, ternyata tim B menang dengan jumlah kartu 9, maka skor menjadi 0-1. Untuk nilai besar didapat dari nilai kecil jika telah mencapai 30.

### III. ALGORITMA GREEDY

Algoritma greedy adalah salah satu algoritma pemecahan masalah yang cukup banyak digunakan. Algoritma ini cukup simple. Algoritma ini lebih mangkus dan lebih 'cerdas' dibandingkan dengan algoritma brute force. Arti greedy sendiri adalah tamak, rakus atau loba. Semboyan algoritma ini adalah : "*Take what you can get now!*".

Algoritma greedy mempunyai karakteristik / prinsip optimalitas. Pada setiap langkah, algoritma greedy memilih nilai optimal lokal (maksimum lokal atau minimum lokal). Diharapkan dengan memilih nilai optimal lokal, maka akan terbentuk nilai optimal global.

Elemen-elemen algoritma greedy adalah :

#### 1. Himpunan kandidat

Merupakan himpunan yang berisi elemen – elemen pembentuk solusi. Pada setiap langkah, satu buah kandidat diambil dari himpunannya. Contohnya adalah himpunan simpul dalam graf untuk menentukan graf merentang minimum.

2. **Himpunan solusi**  
Merupakan himpunan yang berisi elemen – elemen pembentuk solusi. Pada setiap langkah, satu buah kandidat diambil dari himpunannya. Contohnya adalah himpunan simpul dalam graf untuk menentukan graf merentang minimum.
3. **Fungsi seleksi**  
Fungsi yang pada setiap langkah memilih kandidat yang paling memungkinkan mencapai solusi optimal. Kandidat yang sudah dipilih pada suatu langkah tidak akan pernah dipertimbangkan lagi pada langkah selanjutnya.
4. **Fungsi kelayakan**  
Fungsi yang memeriksa apakah suatu kandidat yang dipilih dapat memberikan solusi yang layak, dimana kandidat tersebut dan himpunan solusi yang terbentuk tidak melanggar kendala/*constraint* yang ada.
5. **Fungsi objektif** : fungsi yang memaksimalkan atau meminimumkan nilai solusi

Dengan kata lain, algoritma *greedy* melibatkan pencarian sebuah himpunan bagian  $S$ , dari himpunan kandidat,  $C$ ; yang dalam hal ini,  $S$  harus memenuhi beberapa kriteria yang ditentukan, yaitu menyatakan suatu solusi dan  $S$  dioptimisasi oleh fungsi obyektif.

Algoritma ini lebih mangkus dibandingkan algoritma brute force (exhaustive search), namun algoritma greedy sendiri bukan algoritma yang paling mangkus dalam menghasilkan nilai optimum, karena fungsi seleksinya hanya memilih optimum lokal, yang dimana optimum lokal tersebut belum tentu himpunan bagian dari solusi optimum global.

Algoritma greedy sering digunakan untuk mendapatkan nilai/solusi hampiran (approximation value) jika nilai terbaik tidak mutlak diperlukan.

**Kelebihan :**

1. Lebih mangkus dan lebih ‘cerdas’ dibandingkan algoritma brute force.
2. Cukup sederhana untuk menghasilkan nilai optimum yang mendekati atau bahkan bisa sama, jika dibandingkan dengan algoritma yang lebih kompleks
3. Dapat memberikan nilai/solusi hampiran yang cukup baik untuk kompleksitas algoritma yang jauh lebih rendah dibandingkan exhaustive search.

**Kekurangan :**

1. Algoritma greedy tidak beroperasi secara menyeluruh terhadap seluruh kemungkinan solusi yang ada, sehingga tidak dapat dipastikan bahwa hasilnya adalah solusi optimum terbaik

2. Terdapat beberapa fungsi seleksi yang berbeda sehingga kita harus memilih fungsi seleksi yang tepat untuk menghasilkan solusi yang paling optimum
3. Akibat dari kedua hal diatas, maka algoritma greedy tidak selalu berhasil memberikan solusi terbaik.

#### IV. PENERAPAN ALGORITMA GREEDY

##### A. Algoritma greedy untuk melakukan penawaran

Penawaran pada awal permainan dimulai dari orang/rekanan yang mengkokok kartu. Setelah permainan berjalan, penawaran dimulai dari tim yang memenangkan permainan sebelumnya.

1. Himpunan kandidat  
Himpunan kandidat dalam persoalan permainan truf pass ini adalah 13 buah kartu yang dibagikan secara acak dari satu set kartu remi tanpa joker
2. Himpunan solusi  
Himpunan solusi dalam persoalan ini adalah bunga kartu yang menjadi truf dan
3. Fungsi seleksi  
Fungsi untuk menghitung jumlah kartu bernilai tinggi, dan menghitung jumlah kartu tiap bunga
4. Fungsi kelayakan  
Fungsi untuk memeriksa kartu penawaran. Memilih bunga yang mempunyai status ‘terkuat’, yaitu terdapat kartu tinggi, misalnya, terdapat As, Queen, Jack, 10. Atau terdapat cukup banyak kartu dalam bunga itu, contoh terdapat 6 kartu yang bernilai 4, 6, 7, 8, 9, Jack). Jika rekan menawar pada bunga yang sama dengan yang kita tawar atau dengan kartu terkuat kita, lakukan penawaran atas tawaran lawan pada bunga yang sama dengan rekanan kita.
5. Fungsi objektif  
Menghasilkan kartu penawaran dengan jumlah dan bunga tertentu yang terbaik (Kartu tinggi banyak, kartu cukup panjang, atau sesuai dengan penawaran rekan

##### B. Algoritma greedy untuk menurunkan kartu

Jika seorang pemain mempunyai kartu tertinggi pada bunga tersebut, maka pemain tersebut mengeluarkan kartu tertinggi tersebut, jika tidak ada, mengeluarkan kartu terendah. Khusus untuk pemain ketiga mengeluarkan kartu tertinggi jika kedua pemain sebelumnya mengeluarkan kartu yang lebih rendah. Jika bunga kartu telah void, turunkan kartu truf. Untuk orang pertama, turunkan kartu terendah yang ditawarkan , kartu terbesar pada saat bermain atau kartu lain yang tertinggi atau terendah.

### 1. Himpunan kandidat

Himpunan kandidat dalam persoalan permainan truf pass ini adalah 13 buah kartu yang dibagikan secara acak dari satu set kartu remi tanpa joker

### 2. Himpunan solusi

Himpunan solusi dalam persoalan ini adalah kartu terkuat, sehingga kartu tersebut dapat melakukan penarikan

### 3. Fungsi seleksi

Untuk pemain pertama, turunkan kartu terendah pada bunga yang ditawarkan oleh rekanan, atau kartu tertinggi di bunga manapun. Jika tidak memegang kartu tertinggi, maka turunkan kartu terendah di bunga manapun.

Untuk pemain kedua, turunkan kartu tertinggi dalam permainan berjalan yang dimainkan oleh pemain pertama. Jika tidak ada turunkan kartu terendah.

Untuk pemain ketiga, jika rekanan menurunkan kartu tertinggi, maka turunkan kartu terendah. Jika tidak, turunkan kartu yang tertinggi yang dimiliki walaupun bukan kartu tertinggi dalam permainan.

Untuk pemain keempat, jika rekanan (pemain kedua) atau pemain lawan menurunkan kartu tertinggi, maka turunkan kartu terendah. Jika tidak, turunkan kartu yang lebih tinggi daripada kartu yang sudah diturunkan

### 4. Fungsi kelayakan

Kartu yang diturunkan merupakan kartu tertinggi sehingga dapat melakukan penarikan.

### 5. Fungsi objektif

Jumlah kartu yang ditarik lebih dari sama dengan penawaran.

## C. Implementasi Algoritma greedy untuk melakukan penawaran dalam pseudocode

```
function tawar (Set kartu,
penawaran penawaranTerakhir,
penawaran penawaranTeman) ->
Penawaran
    //Kamus data Lokal
    hasilTawar : Penawaran
    kartuTerkuat : Kartu
    kartuTerpanjang : Kartu ;

    kartuTerkuat <-
    temukanKartuTerkuat(kartu)
    //menghasilkan kartu terkuat yang
    dimiliki

    kartuTerpanjang <-
    temukanKartuTerpanjang(kartu) //
    menghasilkan kartu yang jumlahnya
    terbanyak
```

```
    if (penawaran terakhir = nil)
then
    hasilTawar.jumlah<-7
    hasilTawar.kartu<-
    kartuTerkuat;
    else if (penawaranTeman=nil)
then
    if (kartuTerkuat lebih
tinggi penawaranTerakhir.kartu) then
    hasilTawar.jumlah<-
    penawaranTerakhir.jumlah;
    hasilTawar.kartu<-
    kartuTerkuat
    else
    hasilTawar.jumlah<-
    penawaranTerakhir.jumlah;
    hasilTawar.kartu<-
    kartuTerkuat
    else
    if (penawaranTeman.kartu =
kartuTerkuat) then
    if (kartuTerkuat
lebih tinggi
penawaranTerakhir.kartu) then

    hasilTawar.jumlah<-
    penawaranTerakhir.jumlah;

    hasilTawar.kartu<-kartuTerkuat
    else

    hasilTawar.jumlah<-
    penawaranTerakhir.jumlah;

    hasilTawar.kartu<-kartuTerkuat
    if (penawaranTeman.kartu =
kartuTerpanjang) then
    if (kartuTerpanjang
lebih tinggi
penawaranTerakhir.kartu) then

    hasilTawar.jumlah<-
    penawaranTerakhir.jumlah;

    hasilTawar.kartu<-
    kartuTerpanjang
    else

    hasilTawar.jumlah<-
    penawaranTerakhir.jumlah;

    hasilTawar.kartu<-
    kartuTerpanjang
    return hasilTawar
```

#### D. Implementasi Algoritma greedy untuk menurunkan kartu dalam pseudocode

```
function adaKartuLebihTinggi() -> boolean
//terdapat kartu yang lebih tinggi daripada kartu yang sudah diturunkan pemain sebelumnya
function adaKartuTertinggi (Set kartuKeluar) -> boolean {
// menghasilkan nilai true jika terdapat kartu tertinggi pada kartu yang sedang kita miliki
// set kartuKeluar berisi kartu yang telah diturunkan oleh pemain
// set kartuKeluar diisi ketika keempat pemain sudah menurunkan kartunya
// contoh : jika As sudah terdapat di set KartuKeluar maka King adalah tertinggi
}

function turunkanKartuTertinggi() -> Kartu
Return (HIGHEST_CARD) // HIGHEST_CARD adalah kartu tertinggi yang dimiliki pemain yang sesuai dengan bunga yang dimainkan
function turunkanKartuTerendah() -> Kartu
Return (LOWEST_CARD) // LOWEST_CARD adalah kartu terendah yang dimiliki pemain yang sesuai dengan bunga yang dimainkan

function turunkanTrufTertinggi ()-> Kartu
Return (HIGHEST_TRUF) // HIGHEST_TRUF adalah truf tertinggi yang dimiliki

function turunkanTrufTerendah()-> Kartu
Return (LOWEST_TRUF)
// LOWEST_TRUF adalah truf tertinggi yang dimiliki

function selectKartu {}
If (urutanPemain = 1) then
if (adaKartuTertinggi())
then

turunkanKartuTertinggi()
else if (kartuVoid) then

turunkanTrufTerkecil()
else

turunkanKartuTerkecil()
```

```
else if (urutanPemain = 2) then
if (adaKartuTertinggi())
then

turunkanKartuTertinggi()
else if (kartuVoid) then

turunkanTrufTerendah()
else

turunkanKartuTerendah()
else if (urutanPemain = 3) then
if
(adaKartuYangLebihTinggi()) then

turunkanKartuTertinggi()
else if (kartuVoid()) then

turunkanTrufTertinggi()
else

turunkanKartuTerendah()
else
if
(adaKartuYangLebihTinggi()) then

turunkanKartuTertinggi()
else if (kartuVoid()) then

turunkanTrufTertinggi()
else

turunkanKartuTerendah()
```

#### IV. ANALISIS

Permainan kartu truf pass adalah permainan yang mengutamakan strategi. Penerapan algoritma greedy dalam permainan ini cukup baik, namun terkadang tidak memberikan hasil yang diinginkan/ hasil optimal. Karena algoritma ini tidak memperhitungkan elemen-elemen lain, seperti kartu apa yang akan dikeluarkan pemain lain.

Algoritma greedy hanya salah satu cara pendekatan untuk menyelesaikan permainan kartu truf pass. Masih banyak pendekatan lain yang dapat dilakukan untuk menyelesaikan permainan truf pass. Mungkin jika pendekatan/algoritma lain tersebut digabung bersama dengan greedy akan bisa memberikan hasil yang cukup optimal

Jika hanya menggunakan algoritma greedy, pemain belum tentu selalu memenangkan permainan. Banyak faktor lain, seperti hasil pembagian kartu yang mungkin pemain hanya mendapat satu buah kartu gambar dan kartunya merata (kartunya paling banyak 4 pada tiap bunga), dan juga faktor keberuntungan dalam menurunkan truf.

## V. KESIMPULAN

Algoritma greedy pada permainan truf dapat digunakan dan kemungkinan menghasilkan solusi yang baik (memenangkan permainan) jika hasil pembagian kartu cukup bagus dan mendapat kartu tinggi cukup banyak.

Algoritma greedy juga dapat dikembangkan secara lebih lanjut supaya bisa memberikan solusi yang lebih optimal. Dan mungkin dari algoritma ini dapat dikembangkan kecerdasan buatan (artificial intelligence) untuk permainan kartu truf sehingga dapat dikembangkan permainan truf pada device seperti Android, iPhone, Windows Phone, dll.

## VI. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih terutama kepada Tuhan Yesus Kristus karena berkat anugerah yang diberikan-Nya makalah ini dapat diselesaikan. Penulis juga mengucapkan terima kasih kepada Bapak Ir. Rinaldi Munir, M.T. selaku dosen pengajar kuliah IF3051 Strategi Algoritma karena berkat kuliah yang diberikan dan buku diktat yang ditulis oleh beliau makalah ini dapat dibuat dan disempurnakan.

## VII DAFTAR PUSTAKA

- [1] Rinaldi Munir, "Diktat Kuliah Strategi Algoritmik", Program Studi Teknik Informatika ITB, 2006
- [2] Anany Levitin, "Design and Analysis of Algorithm". Pearson Education Inc. 2012
- [3] Nikolaus Indra, Makalah "Penerapan Algoritma Greedy pada Permainan Kartu Truf", Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung, 2010

## 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, 02 Mei 2015



Dimpos A.G. Sitorus