

Implementasi Algoritma *Greedy* pada Permainan Kartu Hearts

Irfan Ihsanul Amal - 13516064

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

irfan.ihsanulamal@gmail.com

Abstract—Algoritma *Greedy* merupakan metode yang paling populer untuk memecahkan persoalan optimasi. Prinsip dari algoritma ini yaitu “Take what you can get now!” yang artinya ambil apa yang bisa kamu ambil sekarang. Permainan kartu Hearts adalah permainan kartu klasik yang sering kita temui di Windows XP dan Windows 7 bersamaan dengan game kartu lainnya seperti Freecell, Solitaire, dan Spider Solitaire. Tujuan dari game ini adalah mendapat skor terkecil di akhir game.

Keywords—*greedy, hearts, mimimum, queen spade*

I. PENDAHULUAN

Permainan kartu Hearts adalah permainan kartu dimana pemain harus mendapat skor terkecil untuk dapat memenangkan permainan ini. Permainan berakhir ketika salah satu pemain mencapai skor tertinggi yang sudah disepakati bersama.

Permainan ini pertama kali diperkenalkan pada Windows 3.1 pada tahun 1992. Pada versi berikutnya permainan hearts ini terus disertakan. Hal ini terus dilakukan hingga yang terakhir adalah Windows 7. Awalnya permainan ini dapat dilakukan secara multiplayer online melalui versi networknya. Namun pada versi Windows XP, fitur ini dihapus sehingga permainan ini hanya bisa dimainkan offline dan hanya melawan AI saja. Pada versi selanjutnya yaitu Windows 8 ataupun Windows 10 permainan kartu hearts ini sudah tidak disertakan lagi.

Permainan ini dapat dimainkan oleh empat orang pemain, tiga orang pemain, atau lima orang pemain. Untuk empat orang pemain setiap orang pemain memegang tiga belas kartu. Untuk tiga orang pemain kartu 2 diamond harus dibuang dan setiap pemain memegang tujuh belas kartu. Untuk lima orang pemain kartu 2 diamond dan 2 club harus dibuang dan setiap pemain memegang sepuluh kartu.

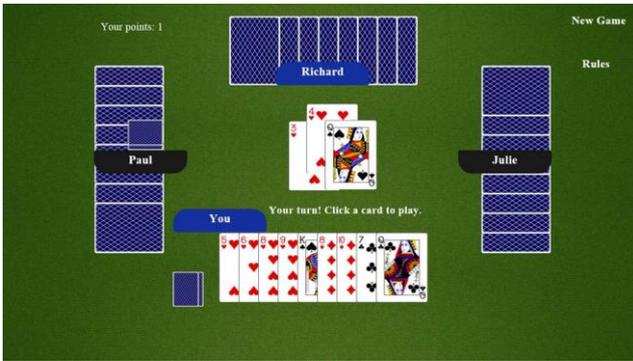
Permainan dimulai dengan melakukan penukaran tiga buah kartu antar pemain terlebih dulu. Kemudian pemain yang memegang 2 club jalan terlebih dulu dengan mengeluarkan kartu tersebut. Jika kartu 2 club sudah dibuang seperti pada permainan lima orang maka kartu 3 club yang dipilih. Dalam setiap round permainan berakhir setelah semua kartu ditangan pemain habis. Setiap giliran setelah 2 atau 3 club dikeluarkan dalam setiap round dimulai oleh pemain yang mengeluarkan

kartu terbesar di giliran sebelumnya (As merupakan kartu terbesar).

Pada setiap giliran pemain harus mengikuti jenis kartu yang orang pertama keluaran (spade, diamond, club, atau heart). Jika tidak ada maka pemain dapat mengeluarkan kartu dengan jenis yang berbeda dan tidak akan dipertimbangkan sebagai kartu yang paling besar. Kartu heart tidak dapat dikeluarkan oleh orang pertama dalam giliran tersebut jika sebelumnya belum ada yang mengeluarkan heart. Kartu heart dapat dikeluarkan jika tidak ada kartu lain di tangan selain kartu heart atau jika pemain tidak memiliki jenis kartu yang dikeluarkan oleh pemain pertama digiliran tersebut. Pemain yang mengeluarkan kartu terbesar akan mengambil semua kartu yang dikeluarkan oleh seluruh pemain untuk penghitungan skor nantinya.

Perhitungan skor dilakukan setiap akhir round dan diakumulasikan setiap roundnya hingga ada pemain yang mencapai batas skor tertinggi yang telah disepakati (Umumnya 100). Peraturan perhitungannya yaitu untuk setiap kartu heart yang diperoleh pada round tersebut akan menambahkan skor sebanyak 1 poin, dan untuk queen spade akan mendapat 13 poin. Namun jika ada pemain yang mendapat seluruh kartu heart dan queen spade, dia tidak akan mendapatkan poin sebanyak 26 melainkan 0 poin, dan semua pemain lain yang akan mendapat 26 poin.

Pada makalah ini akan dijelaskan bagaimana memenangkan permainan kartu ini dengan menggunakan algoritma *greedy*. Strategi penyelesaian akan dibahas berdasarkan case yang mungkin terjadi dalam permainan ini. Penjelasan tidak dilakukan berdasarkan urutan langkah dari awal permainan hingga akhir permainan.



Gambar 1 Gameplay Hearts (Sumber : <https://www.microsoft.com/en-us/store/p/classic-hearts/9wzdnrcrdsnmk#>)

II. DASAR TEORI

Algoritma greedy merupakan metode yang paling populer untuk memecahkan persoalan optimasi. Greedy artinya rakus, tamak. Prinsip greedy yaitu “take what you can get now!” yang artinya “ambil apa yang bisa kamu ambil sekarang!”. Algoritma greedy membentuk solusi langkah per langkah. Pada setiap langkah, terdapat banyak pilihan untuk dievaluasi. Oleh karena itu, pada setiap langkah harus dibuat keputusan yang terbaik dalam menentukan pilihan. Pada setiap langkah, akan didapatkan optimum lokal dengan harapan bahwa langkah sisanya mengarah ke solusi optimum global.

Algoritma greedy adalah algoritma yang memecahkan masalah langkah per langkah. Pada setiap langkah, pilihan terbaik yang diperoleh pada saat itu akan diambil tanpa memperhatikan konsekuensi ke depan. Dengan memilih optimum lokal pada setiap langkah diharapkan akan berakhir dengan optimum global.

Elemen-elemen algoritma greedy:

1. Himpunan Kandidat (C)
Himpunan kandidat adalah himpunan semua elemen yang dapat dipilih sebagai solusi.
2. Himpunan Solusi (S)
Himpunan solusi adalah himpunan yang berisi semua kandidat yang dipilih sebagai solusi.
3. Fungsi Seleksi
Fungsi seleksi adalah fungsi untuk memilih kandidat yang paling optimal untuk dijadikan solusi.
4. Fungsi Kelayakan
Fungsi kelayakan adalah fungsi untuk memeriksa hasil dari fungsi seleksi, apakah memenuhi constraint yang ada atau tidak.
5. Fungsi Obyektif
Fungsi obyektif adalah fungsi untuk mengoptimalkan nilai solusi.

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 dioptimasi oleh fungsi obyektif.

Secara umum skema algoritma greedy adalah sebagai berikut:

```
function greedy(input C: himpunan_kandidat) → himpunan_kandidat
  ( Mengembalikan solusi dari persoalan optimasi dengan algoritma greedy
  Masukan: himpunan_kandidat C
  Keluaran: himpunan_solusi yang bertipe himpunan_kandidat
  )
  Deklarasi
    x : kandidat
    S : himpunan_kandidat
  Algoritma:
    S ← {} ( inisialisasi S dengan kosong )
    while (not SOLUSI(S)) and (C ≠ {}) do
      x ← SELEKSI(C) ( pilih sebuah kandidat dari C )
      C ← C - {x} ( elemen himpunan_kandidat berkurang satu )
      if LAYAK(S ∪ {x}) then
        S ← S ∪ {x}
      endif
    endwhile
    (SOLUSI(S) or C = {} )
    if SOLUSI(S) then
      return S
    else
      write('tidak ada solusi')
    endif
  endif
```

Optimum global tidak menjamin solusi merupakan optimum. Hal ini dikarenakan algoritma greedy tidak beroperasi secara menyeluruh terhadap semua operasi yang ada. Sehingga algoritma greedy tidak selalu menghasilkan solusi yang optimum.

Jika jawaban mutlak tidak diperlukan, maka algoritma greedy dapat digunakan untuk mendapatkan solusi hampiran, dibanding menggunakan algoritma yang rumit untuk mendapat solusi yang eksak. Namun bila algoritma greedy menghasilkan solusi yang optimum, maka keoptimalannya dapat dibuktikan secara matematis.

Beberapa aplikasi algoritma greedy:

1. Masalah Penukaran Uang
Menentukan koin koin dengan jumlah nilai tertentu dengan jumlah koin minimum.
2. Penjadwalan
Menentukan urutan jadwal pelayanan pelanggan yang meminimumkan waktu di dalam sistem.
3. Activity Selection Problem
Memilih sebanyak mungkin aktivitas yang dapat dilayani
4. Integer Knapsack
Memilih objek yang dapat menghasilkan profit tertinggi dengan constraint berat.
5. Job Scheduling with Deadlines
Memilih job yang menghasilkan keuntungan maksimum dengan constraint deadline.
6. Pohon Merentang Minimum
Membentuk pohon dengan bobot total minimum dari sebuah graf

7. Shortest Path
Menentukan lintasan terpendek dari suatu simpul ke simpul yang lain
8. Pemampatan Data dengan Algoritma Huffman
9. Pecahan Mesir
Memecah suatu nilai pecahan menjadi penjumlahan beberapa pecahan dengan bentuk $1/k_n$ dengan jumlah unit minimum.
10. Connecting Wires
Menghubungkan n titik putih dengan n titik hitam dengan panjang kawat minimum.
11. Bot Battleship
Memilih spot yang paling mungkin ditempati kapal musuh.

III. IMPLEMENTASI

Pada kasus yang akan dibahas, terdapat beberapa asumsi yaitu:

1. Permainan yang dilakukan adalah permainan dengan jumlah empat pemain.
2. Strategi penukaran kartu di awal round tidak akan dibahas.
3. Strategi perolehan skor khusus yaitu dengan memdapat semua kartu heart dan queen spade tidak akan dibahas.
4. Strategi yang dibahas yaitu strategi untuk permainan normal yaitu dengan memperoleh skor terkecil.
5. Strategi yang dibahas terbatas untuk satu round saja karena setiap round akan sama saja.
6. Lawan memilih kartu secara random atau tidak dapat kita tentukan.
7. Rotasi dalam satu giliran adalah searah jarum jam.

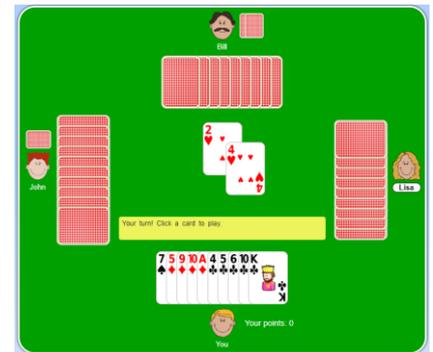
Elemen algoritma greedy pada permasalahan optimasi perolehan skor pada permainan hearts adalah sebagai berikut:

1. Himpunan Kandidat: semua kartu yang lebih kecil dari kartu terbesar yang ada di meja.
2. Himpunan Solusi: Urutan kartu yang dipilih/dikeluarkan hingga habis.
3. Fungsi Seleksi: Kartu yang dipilih adalah kartu yang paling besar dari himpunan kandidat jika harus berjenis sama. Jika bisa berbeda jenis pilih kartu terbesar.
4. Fungsi Kelayakan: Kartu yang dipilih adalah kartu yang sama jenisnya dengan kartu yang pertama ada di meja. Jika tidak ada maka bebas memilih kartu apapun.

5. Fungsi Obyektif: Skor yang didapatkan dalam satu round minimum.

Setelah proses penukaran kartu antar pemain sudah dilakukan, maka pemain dengan kartu 2 club jalan terlebih dulu. Selanjutnya sebagai berikut:

1. Lihat kartu di meja, daftar semua kartu berjenis sama dengan kartu pertama yang ada di meja.
 - a. Jika tidak ada, cek apakah ada kartu heart atau queen spade di tangan
 - i. Jika tidak ada cek apakah ada kartu spade yang lebih besar dari queen di tangan
 1. Jika tidak ada, keluarkan kartu terbesar.



Contoh: pada kasus ini pemain tidak memiliki kartu heart ataupun queen spade di tangan dan juga tidak memiliki kartu spade yang lebih besar dari queen. Maka kartu yang dipilih adalah kartu terbesar yaitu As diamond.

2. Jika ada, keluarkan kartu terbesar diantara kartu tersebut (jika ada as dan king keluarkan as).

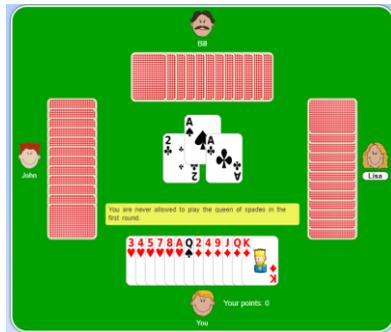


Contoh: pada kasus ini pemain tidak memiliki kartu club, heart ataupun queen spade di tangan namun ada kartu spade yang lebih besar dari queen di tangan yaitu As. Maka kartu yang dipilih adalah As spade.

- ii. Jika ada cek apa bisa dikeluarkan

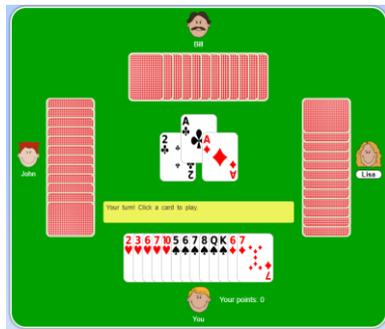
1. Jika tidak bisa, cek apakah ada kartu spade yang lebih besar dari queen di tangan.

a. Jika tidak ada, keluarkan kartu terbesar yang ada di tangan selain heart dan queen spade.



Contoh: pada kasus ini pemain tidak memiliki kartu club di tangan dan pemain belum bisa mengeluarkan heart ataupun queen spade. Maka kartu yang dipilih adalah king diamond.

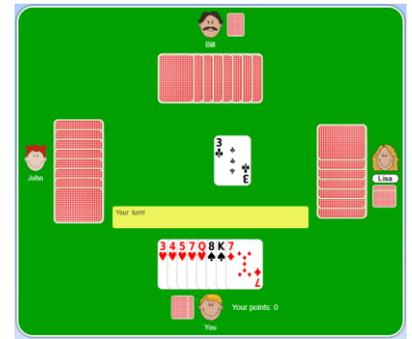
b. Jika ada, keluarkan kartu terbesar diantara kartu tersebut (jika ada as dan king keluarkan as).



Contoh: pada kasus ini pemain tidak memiliki kartu club dan tidak dapat mengeluarkan kartu queen spade ataupun kartu heart. Namun, terdapat kartu yang lebih besar dari queen spade yaitu king spade. Maka pilih kartu king spade.

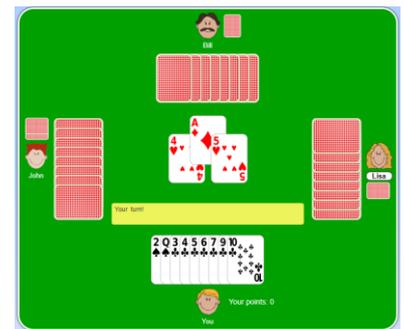
2. Jika bisa, cek apa ada queen spade di tangan.

a. Jika tidak keluarkan kartu heart terbesar yang ada di tangan.



Contoh: pada kasus ini pemain tidak memiliki kartu club di tangan namun memiliki kartu heart di tangan. Maka kartu yang dipilih adalah kartu heart terbesar yaitu queen heart.

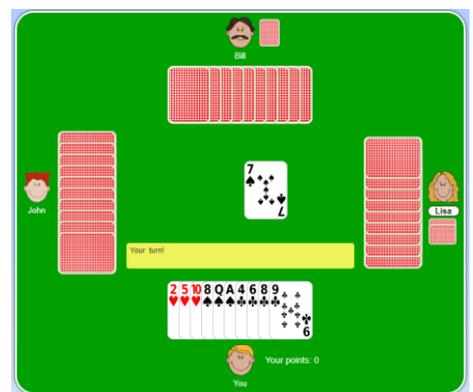
b. Jika ada keluarkan queen spade.



Contoh: pada kasus ini pemain tidak memiliki kartu heart di tangan namun memiliki kartu queen spade di tangan. Maka kartu yang dipilih adalah kartu queen spade.

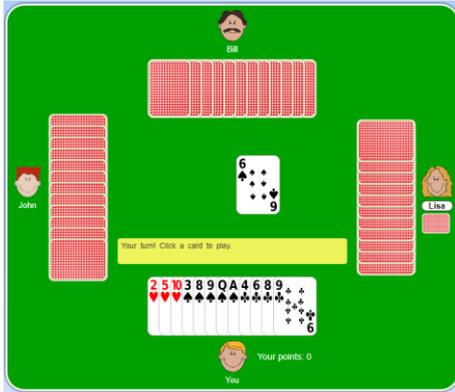
b. Jika ada, daftar semua kartu yang lebih kecil dari kartu terbesar dan berjenis sama di meja

i. Jika tidak ada, pilih kartu terbesar berjenis sama di tangan.



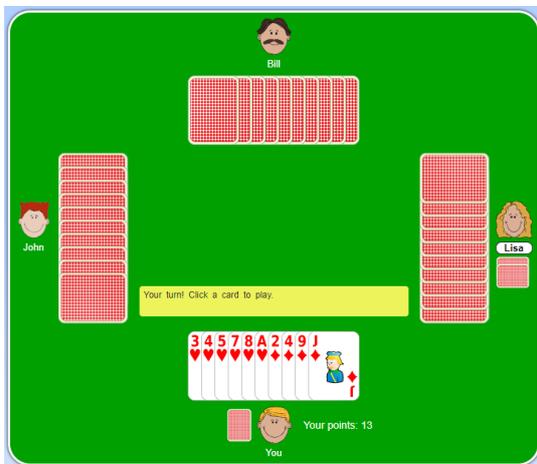
Contoh: pada kasus ini pemain memiliki kartu spade di tangan namun tidak ada kartu yang lebih kecil dari 7. Maka pilih kartu terbesar yaitu As spade.

- ii. Jika ada, pilih kartu yang paling besar diantaranya.



Contoh: pada kasus ini pemain memiliki kartu spade di tangan. Dari tangan didaftar kartu yang lebih kecil dari kartu terbesar berjenis sama di meja dalam hal ini 6 spade. Hasilnya 3 spade. Pilih kartu terbesar dari daftar untuk dikeluarkan yaitu 3 spade.

2. Jika jalan paling awal, pilihlah kartu terkecil di tangan.



Contoh: pada kasus ini pemain jalan paling awal. Maka kartu yang dipilih adalah 2 diamond.

Secara keseluruhan strateginya adalah usahakan keluaran kartu queen spade disaat tidak memiliki kartu dengan jenis yang sama jika tidak ada maka kartu heart terbesar yang dipilih, hal ini bertujuan untuk menghindari akumulasi poin dari kartu tersebut. Selalu keluaran kartu terbesar yang lebih kecil dari kartu berjenis sama di meja, hal ini bertujuan agar jika ada yang mengeluarkan kartu queen spade atau heart kita tidak akan mendapatkannya, namun kita tetap dapat membuang kartu yang cukup besar. Jika tidak ada kartu yang lebih kecil keluaran kartu terbesar, hal ini dilakukan karena kartu manapun akan membuat kita memperoleh poin jika ada, sehingga lebih baik mengeluarkan kartu yang paling besar. Jika jalan paling awal, keluaran kartu terkecil yang ada di tangan, hal ini bertujuan untuk menghindari kita memperoleh poin.

IV. KESIMPULAN

Pada persoalan optimasi algoritma greedy merupakan alternatif pilihan yang paling sering dipilih karena cepat dan mudah. Walaupun solusi yang dihasilkan tidaklah selalu optimum. Namun, solusi tersebut biasanya sudah mendekati atau merupakan hampiran dari solusi optimalnya. Nilai hampiran atau aproksimasi tersebut untuk beberapa kasus biasanya sudah dapat diterima. Hal ini alasan lain yang membuat algoritma greedy menjadi cukup populer dan sering dipakai dalam berbagai masalah optimasi.

Namun, untuk persoalan yang membutuhkan solusi yang optimum dan eksak, algoritma ini sebaiknya tidak dijadikan alternative karena kekurangannya tersebut. Terutama jika penggunaannya merupakan aplikasi atau program yang vital yang dapat menentukan kehidupan orang banyak seperti program yang berkaitan dengan pesawat terbang atau program yang berkaitan dengan reaktor nuklir. Namun jika algoritma ini menghasilkan solusi yang optimum, maka keoptimalannya dapat dibuktikan secara matematis. Dengan kata lain, algoritma greedy belum tentu menghasilkan hasil yang optimum, namun jika solusi hasilnya optimum, maka dapat dibuktikan secara matematis.

Begitu pula untuk permainan kartu hearts ini, solusi optimum tidak selalu dapat tercapai karena ada hal hal yang lain yang harus dipertimbangkan seperti strategi penukaran kartu di awal round dan juga skor khusus dengan memperoleh semua kartu hearts dan queen spade.

Penggunaan algoritma ini sering menghasilkan nilai optimum yaitu dengan menghasilkan skor yang terkecil di akhir pertandingan. Namun kadang kala hasilnya hanya berupa hampiran dalam hal ini adalah juara 2 atau terburuk 3. Selama saya memainkan game ini dengan algoritma greedy tersebut saya belum pernah juara ke 4 untuk permainan empat orang.

Untuk mendapat solusi yang pasti optimum biasanya dapat digunakan algoritma brute force seperti exhaustive search. Namun pada permainan ini sepertinya penggunaan algoritma ini sangat sulit karena ada banyak kemungkinan persebaran kartu pada setiap pemainnya dan ada banyak kemungkinan kasus untuk setiap giliran dalam sebuah round. Sehingga cost untuk menggunakan algoritma exhaustive search menjadi sangat besar.

Meskipun algoritma greedy tidak menghasilkan hasil yang selalu optimum pada permainan hearts ini, namun algoritma ini dapat menjadi alternatif yang sangat disarankan karena mudah dan mendekati optimum.

Algoritma greedy ini dapat dikembangkan menjadi lebih rakus lagi terutama pada bagian strategi penukaran kartu di awal round dan strategi skor khusus dengan memperoleh semua kartu hearts dan kartu queen spade, ataupun dengan terus meningkatkan kinerja dari algoritma yang sudah dipaparkan sebelumnya.

UCAPAN TERIMA KASIH

Pertama-tama saya ingin mengucapkan syukur kepada Tuhan Yang Maha Esa, atas berkat rahmat-Nya saya dapat menyelesaikan makalah ini. Saya juga berterima kasih kepada

Ibu Masayu Leylia Khodra yang telah memberikan materi mengenai algoritma greedy ini, juga pada Bapak Rinaldi Munir yang telah mempersiapkan slide juga diktat yang berisi materi algoritma greedy ini. Selain itu, saya juga ingin mengucapkan terima kasih kepada orang tua dan segala pihak yang telah membantu dalam penyelesaian makalah ini.

REFERENSI

- [1] <https://www.bicyclecards.com/how-to-play/hearts/>
- [2] Munir, Rinaldi. Diktat Kuliah IF2211 Strategi Algoritma. Bandung: Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung, 2009.
- [3] Munir, Rinaldi. 2017. *Bahan Kuliah ke-3 IF2211 Strategi Algoritma Algoritma Greedy*. Bandung: Institut Teknologi Bandung.
- [4] <https://cardgames.io/hearts/>
- [5] <https://www.microsoft.com/en-us/store/p/hearts-card-game/9nblggh6h9k3>
- [6] <https://www.microsoft.com/en-us/store/p/hearts-deluxe/9wzdnrcfj3wl>

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, 29 April 2012
Ttd



Irfan Ihsanul Amal
13516064