

# Penerapan Algoritma Greedy untuk Memaksimalkan Kombo Pilihan dalam Permainan *Chess Rush*

Arif Rahman Amrul Ghani 13518023<sup>1</sup>

Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung  
<sup>1</sup>13518023@std.stei.itb.ac.id

**Abstract**—Algoritma merupakan salah satu hal penting yang digunakan dalam pengembangan perangkat lunak. Salah satu algoritma yang sering digunakan adalah algoritma greedy. Algoritma ini digunakan untuk permasalahan optimasi, yaitu permasalahan yang membutuhkan solusi yang paling optimal, baik itu maksimasi maupun minimasi. Dalam permainan (*game*), algoritma greedy bisa dimanfaatkan untuk optimasi game tersebut, terlebih lagi game yang bergenre strategi. Algoritma greedy bisa digunakan untuk permainan strategi *auto battler* seperti *Chess Rush* guna memperbesar peluang memenangkan permainan tersebut.

**Keywords**—*algoritma, greedy, hero, kombo, optimasi, permainan*

## I. PENDAHULUAN

Perkembangan teknologi yang sangat pesat sudah tidak bisa dipungkiri lagi. Terlebih teknologi komputer. Berbicara tentang teknologi komputer, pasti tidak akan luput dengan yang namanya algoritma. Algoritma seringkali digunakan oleh para pengembang (*developer*) perangkat lunak untuk memudahkan merancang sebuah program. Algoritma mempunyai artian langkah-langkah yang ditempuh untuk menyelesaikan suatu permasalahan.

Perkembangan teknologi ini juga merambah ke dunia hiburan. Permainan menggunakan teknologi komputer / smartphone kini sudah menjamur di khalayak ramai. Sebut saja *Game*. Sekarang *game* sudah bisa dimainkan secara online sehingga para pemainnya dapat bermain bersama dan berinteraksi satu sama lain tanpa tatap muka. Terlebih lagi, *e-sport* sekarang digalakkan di berbagai belahan dunia. Hal ini meningkatkan eksistensi dari *game online*. Selain itu, ini juga memicu para *game developer* untuk membuat *game* yang bagus dan bisa diterima oleh masyarakat. Kini muncul berbagai genre *game*, antara lain: *Adventure, Strategi, Puzzle*, dan masih banyak lagi.

## II. DASAR TEORI

### A. Algoritma Greedy

Algoritma Greedy merupakan salah satu algoritma yang sering digunakan untuk memecahkan suatu persoalan,

terutama persoalan optimasi. Persoalan optimasi dapat diartikan sebagai pencarian solusi terbaik (optimal) dalam suatu permasalahan. Terdapat dua macam persoalan optimasi, yaitu maksimasi (*maximization*) dan minimasi (*minimization*). Contoh persoalan maksimasi adalah *knapsack problem*, suatu permasalahan optimasi dengan tujuan untuk mendapatkan keuntungan sebesar-besarnya dengan cara memasukkan objek-objek ke dalam *knapsack* tanpa melebihi batas muatan *knapsack* itu sendiri. Sedangkan contoh permasalahan minimasi adalah masalah penukaran uang koin. Permasalahan ini menukar sejumlah uang dengan koin-koin yang disediakan. Tujuan dari persoalan ini adalah mendapatkan banyak koin paling sedikit dari koin-koin yang tersedia dan apabila nominalnya dijumlahkan akan sama dengan sejumlah uang yang akan dicari pecahannya.

Prinsip kerja algoritma greedy adalah “*take what can you get now*”. Algoritma ini menggunakan pendekatan penyelesaian masalah dengan mencari nilai terbaik (optimal) dari setiap langkahnya (*step-by-step*) tanpa memperhatikan konsekuensi yang didapat pada langkah selanjutnya. Dengan kata lain, algoritma ini berusaha untuk mencari solusi *local optimum* (nilai solusi terbaik dalam satu langkah) dengan harapan akan mendapatkan solusi *global optimum* (nilai solusi terbaik dalam suatu permasalahan). Dalam pengaplikasiannya, algoritma ini belum tentu memperoleh hasil akhir *global optimum*, karena sering kali terdapat suatu persoalan yang solusi awalnya kurang optimal, namun apabila ditinjau dari keseluruhan hasil akan memberikan hasil yang optimal. Ini merupakan salah satu kekurangan algoritma greedy. Meskipun demikian, algoritma ini memberikan solusi yang mendekati nilai optimal dengan waktu yang lebih cepat. Algoritma ini cocok untuk menyelesaikan suatu permasalahan dengan waktu yang singkat.

Terdapat beberapa elemen dalam algoritma greedy, antara lain:

#### 1. Himpunan Kandidat

Himpunan ini berisi kandidat-kandidat yang akan dicoba satu persatu oleh algoritma greedy untuk mendapatkan hasil yang optimum.

## 2. Himpunan Solusi

Himpunan ini berisi hasil akhir/solusi dari himpunan kandidat yang terpilih.

## 3. Fungsi Seleksi

Fungsi ini menyeleksi himpunan kandidat sebelum dimasukkan ke dalam himpunan solusi.

## 4. Fungsi Kelayakan

Kandidat yang lolos dari fungsi seleksi, akan diperiksa menggunakan fungsi ini untuk menentukan layak tidaknya kandidat tersebut menjadi solusi (memberikan hasil yang diinginkan).

## 5. Fungsi Objektif

Fungsi yang digunakan untuk memaksimalkan atau meminimumkan nilai solusi.

### B. Permainan Chess Rush

Chess Rush (<https://www.chessrushgame.com/>) merupakan salah satu game strategi *auto battler* yang dikeluarkan oleh *game developer* dari Tiongkok, yaitu Tencent. Game ini dirilis tanggal 4 Juli 2019 pada perangkat berbasis Android dan iOS. Sejak awal perilisannya hingga hari ini, Chess Rush sudah didownload oleh lebih dari 5 juta pengguna.



Gambar 1. Tampilan Home

(Sumber : Permainan Chess Rush)

Permainan ini mengadu strategi para pemain dalam menyusun kombo dan posisi hero dalam map berukuran 8x8. Fokus dari permainan ini adalah kemampuan bertahan tiap pemain dalam menghadapi monster atau strategi pemain lain hingga menyisakan satu orang pemain yang menjadi pemenangnya. Permainan ini terdiri dari beberapa ronde, tiap ronde pemain akan melawan 1 pemain lawan (1v1). Apabila pemain kalah, maka HP pemain akan berkurang, apabila menang, maka HP pemain akan tetap. Permainan selesai saat hanya terdapat 1 pemain yang masih bertahan (HP lebih dari 0). Chess Rush menggunakan tampilan 3 dimensi sehingga menambah daya tarik dari game ini.

Permainan ini juga termasuk permainan *seasonal* yaitu permainan yang menggunakan *season* (musim) dalam

beberapa waktu. Season pada game ini bertujuan untuk pemeringkatan para pemain di seluruh dunia. Peringkat (*Tier*) yang tersedia dalam game ini antara lain:

1. Tidak ada peringkat
2. Bronze 1-10
3. Silver 1-10
4. Gold 1-10
5. Platinum 1-10
6. Diamond 1-10
7. Master

Setiap pergantian season, maka *tier* semua pemain akan diatur ulang sehingga tidak ada batasan maksimal tier dalam permainan ini. Saat ini, Permainan Chess Rush memasuki season 5 hingga tanggal 17 Juni 2020.

Setiap pemain juga bisa mengatur skin karakter, skin map, hiasan, dan efek pada saat pemunculan hero. Pada gambar 1, pemain menggunakan skin karakter "Roh Monyet" dan skin papan "Padang Rumput Blusterleaf". Skin dapat diperoleh dengan cara membeli menggunakan gold atau voucher. Skin juga bisa diperoleh dengan menyelesaikan misi, atau mendapat hadiah seasonal. Gold bisa diperoleh dengan cara bermain game. Voucher bisa diperoleh dengan cara top-up atau hadiah menyelesaikan misi.



Gambar 2. Map Chess Rush

Map yang digunakan dalam permainan ini berukuran 8x8 dengan pembagian 8x4 untuk pemain dan sisanya untuk pemain lawan. Pemain juga mendapat 8 kotak untuk menyimpan hero yang sudah dibeli namun belum ingin dikeluarkan dalam map.

Salah satu kelebihan dari permainan ini adalah terdapat beragam mode yang dapat dimainkan sehingga mengurangi kebosanan para pemainnya. Mode-mode tersebut antara lain:

### 1. Mode Berperingkat

Dalam mode ini, juara 1-4 mendapat tambahan XP sedangkan juara 5-8 akan mendapat pengurangan XP. Pemain juga akan mendapat tambahan *gold* sesuai dengan peringkatnya masing-masing. Gold bisa digunakan untuk membeli *skin* maupun tambahan lainnya.

#### a. Mode Klasik

*Classic Mode* merupakan mode bermain dasar dengan 8 pemain yang saling beradu strategi (1v1 setiap rondanya), dengan HP tiap pemain adalah 100.

b. Mode Turbo

*Turbo Mode* merupakan duplikasi *Classic Mode* dengan waktu yang lebih cepat dan HP tiap pemain yang lebih sedikit, yaitu 60.

c. Squad Class

*Squad Class* merupakan mode bermain tim, 4v4 (1v1 setiap rondanya) dengan HP akumulasi sebanyak 500 untuk setiap tim. Apabila pemain menang melawan musuh, hero pemain masih tersisa, dan teman satu tim masih dalam peperangan, hero dapat berpindah ke map teman satu tim untuk membantu memenangkan pertarungan dalam satu ronde.

2. Kasual

a. Arena Emas (Tantangan)

Menggunakan gold untuk masuk *room*, mendapatkan gold lebih banyak apabila menang.

b. Arena Emas (Master)

Sama seperti Arena Emas (Tantangan) dengan biaya masuk dan hadiah lebih banyak.

3. Mode Dasar

a. Mode Latihan

Digunakan untuk membiasakan diri dengan *gameplay*-nya, melawan pemain lain, namun apabila kalah tidak akan menurunkan pangkat akun pemain. HP pemain dalam mode ini adalah 30.

b. Tutorial

Mode yang digunakan untuk para pemain pemula yang baru belajar bermain game Chess Rush.

c. Mode Bot

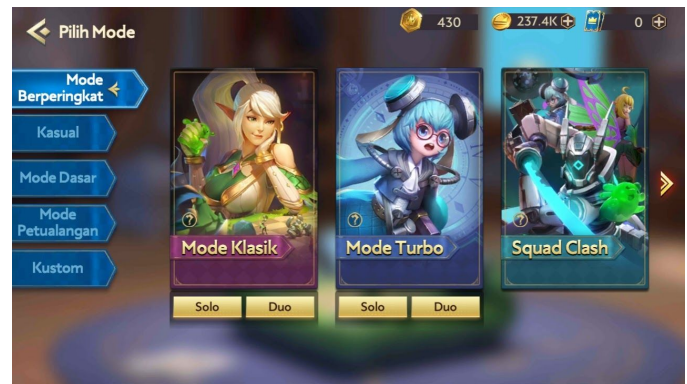
Seperti *Classic Mode*, namun melawan bot.

4. Mode Petualangan

Mode terbaru dalam season 5 Chess Rush.

5. Kustom

Mode ini digunakan untuk pemain agar bisa bermain dengan teman-temannya dengan cara membuat *room* sendiri.



Gambar 3. Pilihan Mode

Selain banyak mode, terdapat banyak hero yang dapat digunakan dalam Chess Rush. Hero-hero tersebut memiliki Class dan Ras masing-masing (bisa lebih dari satu). Setiap Class dan Ras juga memiliki kelebihan masing-masing. Ras dalam Chess Rush adalah sebagai berikut:

1. Eastern : Memberikan efek “*Iron Body*” selama beberapa detik untuk hero Eastern
2. Dragon : Memulihkan mana secara instan di awal ronde untuk hero Dragon
3. Jelly : Menaikkan level hero yang dipilih secara otomatis
4. Goblin : Meningkatkan kecepatan serangan untuk semua hero
5. Beast : Meningkatkan *damage*, menambah pasukan
6. Elf : Menjerat musuh selama beberapa detik sehingga musuh tidak dapat melakukan apapun.
7. Human : Memulihkan mana hero
8. Undead : Mengurangi armor musuh
9. Oceanborn : Pertahanan dari sihir untuk semua hero
10. Mecha : Meningkatkan armor, regenerasi HP hero
11. Void : Mengurangi HP hero musuh secara acak
12. Demon : Menambah *physical damage*
13. Angel : Menyingkirkan satu hero musuh, memulihkan HP hero, meningkatkan damage

Kelas dalam Chess Rush adalah sebagai berikut:

1. Warrior : Meningkatkan Armor
2. Assassin : Menjadi tak terlihat untuk beberapa detik (hero tidak dapat diserang)
3. Sorcerer : Meningkatkan damage
4. Rider : Meningkatkan armor
5. Summoner : Meningkatkan HP hero Summoner
6. Hunter : Meningkatkan kecepatan serangan
7. Warlock : Menyerap HP hero musuh
8. Punisher : Menambah damage
9. Druid : Mempercepat peningkatan level
10. Protector : menyerap damage hero musuh



Gambar 4. Pemilihan Hero pada Ronde 1

Setiap 3 hero yang sama bisa digabungkan menjadi 1 hero dengan level bintang 2. Setiap 3 hero bintang 2 yang sama bisa digabungkan menjadi 1 hero dengan level bintang 3. Pemain diharapkan bisa menyusun strategi untuk membentuk kombo. Kombo merupakan gabungan kelebihan dari beberapa Ras dan Kelas. Kelebihan dari Ras/Kelas akan aktif apabila terdapat beberapa hero dengan ras/kelas yang sama dalam satu map.

Terdapat banyak sekali hero dengan kombinasi Ras dan Kelas masing-masing, antara lain:

#### 1. Kelas Warrior

Terdiri dari Bangar (Beast), Icetusk (Beast), Steel Fist (Mecha), General Puffer (Oceanborn), Gluttony (Undead), Wukong (Eastern), General of War (Human), Berserker (Goblin), dan Diablo (Demon).

#### 2. Kelas Sorcerer

Terdiri dari Miss Supersonic (Mecha), Frost Lotus (Human), Moonlight Dragon (Elf/Dragon), Void Destruction (Void), Flamecaller (Human), Lightbringer (Human), Tong Sam Cong (Eastern), Apocalypse (Undead), dan Joey (Mecha).

#### 3. Kelas Hunter

Terdiri dari Famine (Undead), Hog Rider (Goblin/Rider), Wujing (Eastern), Cerberus (Demon), Forest's Breath (Elf), Stalker Drone S90 (Mecha), Kraken (Oceanborn), dan Shakaraka (Oceanborn).

#### 4. Kelas Assassin

Terdiri dari Bionic Ninja (Mecha), Lilith (Demon), Siren (Oceanborn), Stinger (Elf/Undead), Venom (Dragon/Beast), Horned Menace (Beast), Headreaper (Goblin), dan Lasse (Elf)

#### 5. Kelas Rider

Terdiri dari Hog Rider (Goblin/Hunter), Scourge (Demon), Wolf Cavalier (Elf), Templar Knight (Human), Calamity (Undead), dan Dragoon (Eastern/Dragon).

#### 6. Kelas Summoner

Terdiri dari Jenne D'Arc (Human), Everbloom (Elf/Druid), Marsh Lord (Beast/Warlock), Nightowl (Beast), Dr. Molly (Mecha), dan Michael (Angel).

#### 7. Kelas Warlock

Terdiri dari Voodoo Elder (Goblin), Marsh Lord (Beast/Warlock), Nether Prince (Demon), Misfortune (Undead), dan Nebula (Void).

#### 8. Kelas Punisher

Terdiri dari Nightingale (Elf), Lancelot (Human), Pemakan Jiwa (Demon/Angel), Dimension Walker (Void), dan Nezha (Eastern).

#### 9. Kelas Druid

Terdiri dari Witch Doctor (Beast), Evergreen (Elf), Everbloom (Elf/Summoner), Gurr (Jelly).

#### 10. Kelas Protector

Terdiri dari Pat Kay (Eastern), Protostar (Void), dan Groundsmasher (Mecha)

Terdapat beberapa istilah penting yang digunakan dalam permainan, antara lain:

1. **Mana Crystal** : uang (gold) yang digunakan untuk membeli hero, refresh, dan memperbesar maksimal hero yang bisa diletakkan di map.
2. **Refund** : Penjualan hero kembali, pemain mendapatkan mana crystal sesuai dengan harga hero yang dijual.
3. **Refresh** : Pengacakan ulang 5 hero yang bisa dibeli oleh pemain. Pengacakan ini mengurangi 2 mana crystal pemain.

Secara garis besar, *gameplay* dari permainan Chess Rush adalah sebagai berikut:

1. Pada awal permainan (Ronde Pertama), pemain memiliki 3 Mana Crystal dengan maksimal hero yang dapat diletakkan di map adalah 2. Pemain dapat membeli hero, menyusun hero dalam map, menjual hero, dan menambah jumlah maksimal hero yang bisa diletakkan dalam map saat persiapan (30 detik) sebelum ronde dimulai. Terdapat 5 hero secara acak yang dapat dibeli dengan Mana Crystal. Pilihan tersebut juga dapat di *refresh* untuk mendapatkan hero yang diinginkan.
2. Saat ronde dimulai, pemain hanya bisa melihat hero miliknya bertanding dengan hero pemain lain, membeli hero, meningkatkan level hero, meningkatkan jumlah maksimal hero yang bisa diletakkan di map, dan menjual hero.
3. Tidak ada batasan ronde, permainan akan berakhir apabila menemukan pemenang.
4. Pemain akan melawan 7 pemain lain, setiap ronde melawan 1 pemain lawan secara acak.

5. Apabila pemain kalah saat melawan pemain lain, maka HPnya akan berkurang.
6. Setiap pergantian ronde, pemain akan memperoleh tambahan 5 mana kristal ditambah bonus 10% dari mana crystal saat ini (maksimal 5), dan ditambah bonus apabila melakukan *win streak*. Maksimal mana crystal yang bisa dimiliki oleh setiap pemain adalah 100.
7. Pemenang permainan adalah satu satunya pemain yang masih bertahan (mempunyai HP lebih dari 0).

### III. PEMBAHASAN

Dalam permainan Chess Rush, algoritma greedy digunakan untuk menentukan apa saja yang dilakukan pada persiapan dalam satu ronde agar bisa memenangkan ronde tersebut. Pada saat persiapan, pemain dapat melakukan:

1. Pembelian hero
2. Penjualan hero (*Refund*)
3. *Refresh*
4. Penambahan maksimal hero dalam satu map
5. Peningkatan level hero
6. Menempatkan hero dalam map (tidak dibahas dalam makalah ini)

Elemen algoritma greedy yang diterapkan dalam permainan ini adalah sebagai berikut:

1. Himpunan Kandidat  
Himpunan kandidatnya adalah semua hero yang ada dalam permainan Chess Rush.
2. Himpunan Solusi  
Himpunan solusinya adalah semua hero yang dapat dibeli dan diletakkan dalam map, membentuk kombo sehingga memenangkan ronde.
3. Fungsi Seleksi  
Fungsi seleksinya adalah class dan ras hero yang dipilih sesuai dengan yang diinginkan untuk membentuk suatu kombo.
4. Fungsi Kelayakan  
Fungsi kelayakannya adalah mana crystal cukup untuk membeli hero, slot 8 kotak hero belum penuh, dan jumlah hero yang diletakkan dalam map tidak melebihi batas maksimal.
5. Fungsi Objektif  
Membuat kombo se bagus mungkin sehingga bisa memenangkan ronde.

Secara umum, strategi yang penulis buat merupakan pemilihan langkah terbaik menyesuaikan kondisi yang ada. Strategi ini digunakan untuk memaksimalkan kombo yang diinginkan bisa terwujud sehingga belum tentu memenangkan permainan.

Langkah pertama, tetapkan kombo yang diinginkan. Sebagai contoh, penulis memilih kombo 6 *Human* dan 6 *Sorcerer*. Kombo *Human* memiliki kelebihan yaitu memulihkan mana hero, dan Kombo *Sorcerer* memiliki kelebihan Meningkatkan damage. Susunan hero yang

dibutuhkan adalah : Miss Supersonic (Mecha), Frost Lotus (Human), Void Destruction (Void), Flamecaller (Human), Lightbringer (Human), Joey (Mecha), General of War (Human), Templar Knight (Human), dan Lancelot (Human). Dari kombo tersebut juga diperoleh kombo 2 Mecha, dengan kelebihan meningkatkan armor dan meregenerasi HP hero. Kedua, ubah pengaturan penggabungan hero menjadi Auto, sehingga apabila terdapat tiga hero yang sama akan menjadi 1 hero level bintang 2 secara otomatis dan apabila ada 3 hero level bintang 2 yang sama akan menjadi 1 hero level bintang 3 secara otomatis.



Gambar 5. Ronde Pertama

Langkah selanjutnya, penulis memberikan berbagai kemungkinan yang ada dan pilihan tindakan yang dilakukan sehingga menghasilkan nilai *optimum local*. Kemungkinannya antara lain:

1. Apabila Ronde < 10:
  - a. Apabila slot kosong:
    - i. Apabila ada hero yang sesuai dengan kombo yang diinginkan, maka langkah yang dilakukan adalah Membeli Hero. Setelah membeli hero, apabila map belum terisi penuh (dari batas maksimal), maka letakkan hero tersebut ke map. Apabila map sudah penuh, maka tidak melakukan apa-apa.
    - ii. Apabila tidak ada hero yang sesuai, maka cek map. Apabila map belum terisi penuh, maka beli hero bebas dengan harga  $\leq$  mana crystal. Apabila map sudah penuh, maka tidak melakukan apa-apa.
  - b. Apabila slot tidak kosong dan belum penuh:
    - i. Apabila ada hero yang sesuai dengan kombo yang diinginkan, maka langkah yang dilakukan adalah Membeli Hero. Apabila map belum penuh, maka letakkan hero tersebut ke map. Apabila map sudah penuh, maka tidak melakukan apa-apa.

- ii. Apabila tidak ada hero yang sesuai, maka tidak melakukan apa-apa.
- c. Apabila slot sudah penuh:
  - i. Tidak melakukan apa-apa
- 2. Apabila Ronde  $\geq 10$ :
  - a. Apabila slot belum penuh:
    - i. Apabila ada hero yang sesuai dengan kombo yang diinginkan, maka langkah yang dilakukan adalah Membeli Hero. Apabila map belum penuh, maka letakkan hero tersebut ke map. Apabila map sudah penuh, maka langkah yang dilakukan adalah cek mana crystal. Apabila mana  $\geq 40$ , maka beli exp untuk mencicil memperbesar maksimal hero yang bisa diletakkan di map. Setelah itu kembali ke awal.
    - ii. Apabila tidak ada hero yang sesuai, maka cek mana crystal. Apabila mana crystal  $\geq 50$ , maka lakukan Refresh, lalu kembali ke aturan awal. Apabila  $40 \leq \text{mana crystal} < 50$ , maka beli exp untuk mencicil perbesaran maksimal hero yang bisa diletakkan di map, lalu kembali ke aturan awal.
  - b. Apabila slot sudah penuh:
    - i. Apabila ada hero yang sesuai dengan kombo, maka tekan tombol kunci lalu cek mana crystal. Apabila mana crystal  $\geq 40$ , maka beli perluasan lalu kembali ke aturan awal.
    - ii. Apabila tidak ada hero yang sesuai, maka cek mana crystal. Apabila mana crystal  $\geq 50$ , maka refresh, lalu kembali ke aturan awal. Apabila  $40 \leq \text{mana crystal} < 50$ , maka beli exp untuk perbesaran maksimal hero, lalu kembali ke aturan awal.

Algoritma Greedy yang penulis gunakan (dalam pseudocode) adalah sebagai berikut:

**Function**

```
buyHero()
{Membeli hero yang sesuai dengan kombo}

buyExp()
{Membeli Exp, peningkatan max hero yang dapat diletakkan di map}

getHero() → Boolean
```

```
{Mengembalikan boolean, ada/tidak hero yang
dijual masuk dalam daftar kombo}

getMana() → Integer
{Mengembalikan integer banyaknya Mana Crystal}

putHero()
{Meletakkan hero dalam map}

getSlot() → Integer
{Mengembalikan banyak slot penyimpanan hero yang
terisi}

isMapFull() → Boolean
{Mengembalikan boolean, map sudah penuh atau
belum}

refresh()
{Merefresh hero yang dijual dengan membayar 2
Mana Crystal}

Kamus Data:
play : boolean
inRonde : boolean
totalRonde : integer

Algoritma:
while (play):
  inRonde ← True
  while (inRonde):
    if (totalRonde < 10):
      if (getSlot() == 0):
        if (getHero()):
          buyHero()
          if (isMapFull()):
            //Do Nothing
          else:
            putHero()
        else:
          if (isMapFull()):
            //Do Nothing
          else:
            randomBuy()
            putHero()
      else if (getSlot() != 0 and getSlot <
maxSlot):
        if (getHero()):
          buyHero()
          if (isMapFull()):
            //Do Nothing
          else:
            putHero()
        else:
          //Do Nothing
      else:
        //Do Nothing
    else:
      if (getSlot() < maxSlot):
        if (GetHero()):
          buyHero()
          if (isMapFull()):
            if (getMana()  $\geq 40$ ):
              buyExp()
            else:
              putHero()
          totalRonde ← totalRonde - 1
          inRonde ← False
```

```

else:
    if (getMana() >= 50):
        refresh()
    else if (getMana() < 50 and
getMana() >= 40):
        buyExp()
        totalRonde ← totalRonde - 1
        inRonde ← False
else:
    if (getHero()):
        lockHero()
        if (getMana() >= 40):
            buyExp()
            totalRonde ← totalRonde - 1
            inRonde ← False
        else:
            if (getMana() >= 50):
                refresh()
            else if (getMana() < 50 and
getMana() >= 40):
                buyExp()
                totalRonde ← totalRonde - 1
                inRonde ← False
totalRonde ← totalRonde + 1

```

Keterangan :   Kembali ke aturan awal (looping)

#### IV. KESIMPULAN

Algoritma greedy dapat digunakan untuk pengembangan permainan (*game*). Dalam permainan Chess Rush, algoritma ini digunakan untuk memilih/membeli hero yang paling optimal sehingga membentuk kombo dan memenangkan permainan dalam setiap rondonya. Meskipun algoritma greedy belum tentu memberikan solusi yang optimal, akan tetapi algoritma ini cocok digunakan dalam permainan Chess Rush karena waktu yang diberikan setiap rondonya sangat terbatas.

Kelebihan dari penggunaan algoritma greedy pada permainan Chess Rush adalah sebagai berikut:

1. Kemungkinan menang dalam setiap ronde menjadi lebih besar.
2. Kemungkinan memperoleh *win streak* (Menang secara berturut-turut) lebih besar, sehingga menambah bonus mana crystal pada setiap pergantian ronde.

Kekurangan dari penggunaan algoritma greedy pada permainan Chess Rush adalah sebagai berikut:

1. Hero yang muncul merupakan hasil *random*, sehingga hero yang diinginkan tidak pernah muncul bisa terjadi.
2. Pembahasan belum mencakup peletakan hero dalam map (papan).

#### V. UCAPAN TERIMAKASIH

Pertama, Penulis mengucapkan syukur kepada Allah SWT, sehingga dapat menyelesaikan penulisan makalah Strategi

Algoritma ini. Setelah itu, penulis mengucapkan terima kasih kepada Dr. Nur Ulfa Maulidevi, S.T., M.Sc. selaku dosen pengajar IF2211 Strategi Algoritma pada Program Studi Teknik Informatika Institut Teknologi Bandung (ITB) yang telah menyampaikan materi dengan baik dan semoga ilmunya bisa bermanfaat kedepannya. Kemudian, penulis juga mengucapkan terima kasih kepada orangtua penulis, keluarga, teman, dan semua yang telah membantu secara teknis maupun non teknis sehingga penulisan makalah ini bisa terselesaikan. Semoga makalah ini bisa bermanfaat untuk orang banyak.

#### PRANALA VIDEO YOUTUBE

penjelasan menggunakan video bisa dilihat pada laman : <https://youtu.be/zs43rq2qVCk>

#### REFERENSI

- [1] <https://www.chessrushgame.com/> , diakses pada hari Jumat, 1 Mei 2020.
- [2] <https://www.tokopedia.com/blog/game-cara-bermain-tips-chess-rush/> , diakses pada hari Jumat, 1 Mei 2020.
- [3] <https://www.kabargames.id/game-chess-rush-review/> , diakses pada hari Senin, 4 Mei 2020.
- [4] <https://www.geeksforgeeks.org/greedy-algorithms/> , diakses pada hari Senin, 4 Mei 2020.
- [5] Munir, Rinaldi. 2017. "Algoritma Greedy (2017).pptx". Bandung.

#### 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, 3 Mei 2020



Arif Rahman Amrul Ghani  
13518023