

Tugas Besar I IF2211 Strategi Algoritma
Semester II Tahun 2020/2021
Pemanfaatan Algoritma *Greedy* dalam Aplikasi Permainan “Worms”

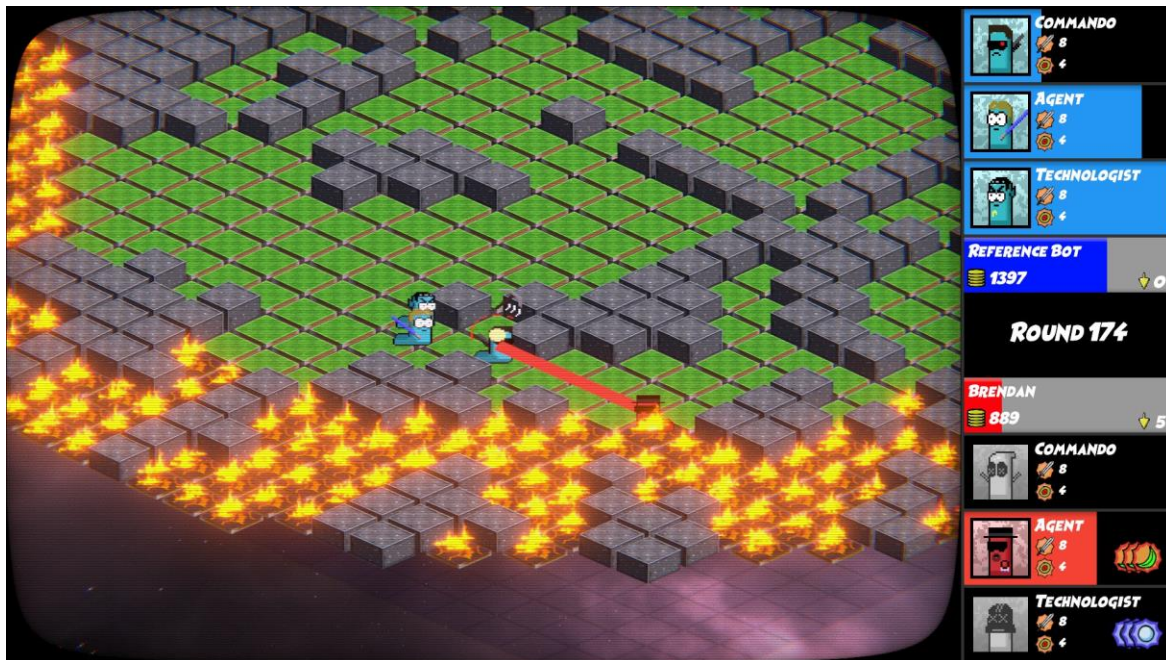
Batas pengumpulan : Jumat, 19 Februari 2021 sebelum pukul 23.59 WIB

Arsip pengumpulan :

- Source program yang bisa dijalankan pada *game engine Worms* disertai *readme.txt*
- Laporan (*soft copy*)

Deskripsi tugas:

Worms adalah sebuah *turned-based game* yang memerlukan strategi untuk memenangkannya. Setiap pemain akan memiliki 3 *worms* dengan perannya masing-masing. Pemain dinyatakan menang jika ia berhasil bertahan hingga akhir permainan dengan cara mengeliminasi pasukan *worms* lawan menggunakan strategi tertentu.



Gambar 1. Contoh tampilan permainan *Worms*

Pada tugas besar pertama Strategi Algoritma ini, gunakanlah sebuah *game engine* untuk mengimplementasikan permainan *Worms*. *Game engine* dapat diperoleh pada laman berikut <https://github.com/EntelectChallenge/2019-Worms>.

Tugas mahasiswa adalah mengimplementasikan seorang “pemain” Worms, dengan menggunakan **strategi greedy** untuk memenangkan permainan. Untuk mengimplementasikan seorang “pemain” tersebut, mahasiswa disarankan melanjutkan program yang terdapat pada *starter bot* di dalam *starter pack* pada laman berikut ini:

(<https://github.com/EntelectChallenge/2019-Worms/releases/tag/2019.3.2>)

Spesifikasi permainan yang digunakan pada tugas besar ini disesuaikan dengan spesifikasi yang disediakan oleh *game engine Worms* pada tautan di atas. Beberapa aturan umum adalah sebagai berikut.

1. Peta permainan berukuran 33x33 *cells*. Terdapat 4 tipe *cell*, yaitu *air*, *dirt*, *deep space*, dan *lava* yang masing-masing memiliki karakteristik berbeda. *Cell* dapat memuat *powerups* yang bisa diambil oleh *worms* yang berada pada *cell* tersebut.
2. Di awal permainan, setiap pemain akan memiliki 3 pasukan *worms* dengan peran dan nilai *health points* yang berbeda, yaitu:
 - a. *Commando*
 - b. *Agent*
 - c. *Technologist*
3. Pada setiap *round*, masing-masing pemain dapat memberikan satu buah *command* untuk pasukan *worm* mereka yang masih aktif (belum tereliminasi). Berikut jenis-jenis *command* yang ada pada permainan:
 - a. *Move*
 - b. *Dig*
 - c. *Shot*
 - d. *Do Nothing*
 - e. *Banana Bomb*
 - f. *Snowball*
 - g. *Select*
4. *Command* dari kedua pemain akan dieksekusi secara bersamaan (bukan sekuensial) dan akan divalidasi terlebih dahulu. *Command* juga akan dieksekusi sesuai urutan prioritas tertentu.
5. Beberapa *command*, seperti *shot* dan *banana bomb* dapat memberikan *damage* pada *worms* target yang terkena serangan, sehingga mengurangi *health points*nya. Jika *health points* suatu *worm* sudah habis, maka *worm* tersebut dinyatakan tereliminasi dari permainan.
6. Permainan akan berakhir ketika salah satu pemain berhasil mengeliminasi seluruh pasukan *worms* lawan atau permainan sudah mencapai jumlah *round* maksimum (400 rounds).

Adapun peraturan yang lebih lengkap dari permainan *Worms*, dapat dilihat pada laman <https://github.com/EntelectChallenge/2019-Worms/blob/develop/game-engine/game-rules.md>.

Spesifikasi tugas:

Pada tugas besar kali ini, anda diminta untuk membuat sebuah *bot* untuk bermain permainan Worms yang telah dijelaskan sebelumnya. Untuk memulai, anda dapat mengikuti panduan singkat sebagai berikut.

1. *Download latest release starter pack.zip* dari tautan berikut. <https://github.com/EntelectChallenge/2019-Worms/releases/tag/2019.3.2>.
2. Untuk menjalankan permainan, kalian butuh beberapa *requirement* dasar sebagai berikut.
 - a. Java (minimal Java 8): <https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>
 - b. IntelliJ IDEA: <https://www.jetbrains.com/idea/>
 - c. NodeJS: <https://nodejs.org/en/download/>
3. Untuk menjalankan permainan, kalian dapat membuka file “run.bat” (Untuk Windows/Mac dapat buka dengan *double-click*, Untuk Linux dapat menjalankan *command* “make run”).
4. Secara *default*, permainan akan dilakukan diantara *reference bot* (*default*-nya berbahasa *JavaScript*) dan starter bot yang disediakan. Untuk mengubah hal tersebut, silahkan edit file “game-runner-config.json”. Anda juga dapat mengubah file “bot.json” untuk mengatur informasi terkait bot anda.
5. Silahkan bersenang-senang dengan memodifikasi bot yang disediakan di *starter-bot*. Ingat bahwa bot kalian harus menggunakan bahasa **Java** dan di-*build* menggunakan **IntelliJ**. **Dilarang** menggunakan kode program tersebut untuk pemainnya atau kode program lain yang diunduh dari Internet. Mahasiswa harus membuat program sendiri, tetapi belajar dari program yang sudah ada tidak dilarang.
6. (Optional) Anda dapat melihat hasil permainan dengan menggunakan *visualizer* berikut <https://github.com/dlweatherhead/entelect-challenge-2019-visualiser/releases/tag/v1.0f1>
7. Untuk referensi lebih lanjut, silahkan eksplorasi di [tautan berikut](#).

Strategi *greedy* yang diimplementasikan tiap kelompok harus dikaitkan dengan fungsi objektif dari permainan itu sendiri, yaitu memenangkan permainan dengan cara mengeliminasi seluruh worms lawan dengan senjata dan skill yang sudah disediakan dalam permainan. Salah satu contoh pendekatan *greedy* yang bisa digunakan (pendekatan tak terbatas pada contoh ini saja) adalah menyerang pasukan lawan dengan senjata dengan *hitpoint* / *damage* terbesar. Buatlah strategi *greedy* terbaik, karena setiap “pemain” dari masing-masing kelompok akan diadu satu sama lain dalam suatu kompetisi Tubes 1 (secara daring).

Strategi *greedy* harus dituliskan secara eksplisit pada laporan, karena akan diperiksa pada saat demo apakah strategi yang dituliskan sesuai dengan yang diimplementasikan. Tiap kelompok dapat menggunakan kreativitas mereka dalam menyusun strategi *greedy* untuk memenangkan permainan. Implementasi pemain harus dapat dijalankan pada game engine yang telah disebutkan pada spesifikasi tugas besar, serta dapat dikompetisikan dengan pemain dari kelompok lain.

Lain-lain:

1. Tugas dikerjakan per kelompok dengan jumlah anggota minimal 2 orang dan maksimal 3 orang. Anggota kelompok diperbolehkan lintas kelas.
2. Program dibuat dalam **Bahasa Java**, sebagai salah satu bahasa yang dapat difasilitasi oleh *game engine* pada Tugas Besar ini. Dengan Java berarti anda sekaligus dapat mempraktekkan konsep pemrograman berorientasi objek (OOP) pada tugas ini.
3. Program harus mengandung komentar yang jelas, dan untuk setiap strategi Greedy yang disebutkan, harus dilengkapi dengan kode sumber yang dibuat.
4. Beri nama pemain anda tersebut dengan nama kelompok anda. Pengaturan konfigurasi nama pemain bisa diatur pada file `bot.json` yang terdapat dalam `starter-bot`.
5. Terdapat contoh implementasi pemain pada *repository* dari *game engine* tersebut, namun mahasiswa dilarang menggunakan kode program tersebut untuk pemainnya atau kode program lain yang diunduh dari Internet. Mahasiswa harus membuat program sendiri, tetapi belajar dari program yang sudah ada tidak dilarang.
6. Mahasiswa dianggap sudah membaca dokumentasi dari *game engine* tersebut, sehingga tidak terjadi kesalahpahaman spesifikasi antara mahasiswa dan asisten.
7. Pengumpulan tugas besar adalah hari **Jumat, 19 Februari 2021** sebelum pukul **23.59 WIB**. Keterlambatan akan mengurangi nilai.
8. **Bonus** (nilai maksimal 10): Setiap kelompok membuat video aplikasi yang mereka buat kemudian mengunggahnya ke YouTube. Video yang dibuat harus memiliki audio dan menampilkan wajah dari setiap anggota kelompok. Pada waktu demo aplikasi di depan asisten, mahasiswa mengakses video YouTube tersebut dan memutarnya di depan asisten. Beberapa contoh video tubes tahun-tahun sebelumnya dapat dilihat di YouTube dengan menggunakan kata kunci "Tubes Stima", "Tugas besar stima", "strategi algoritma", dll.
9. Semua pertanyaan menyangkut tugas ini harus dikomunikasikan melalui QnA yang disediakan agar dapat dicermati oleh semua peserta kuliah IF2211. Laman QnA dapat diakses pada <http://bit.ly/QnATubes1>
10. Demo program menunggu pemberitahuan lebih lanjut dari asisten.
11. Program akan dikompetisikan (diadu) dengan kelompok lain dan disaksikan oleh seluruh peserta kuliah. Pemenang kompetisi akan diberi hadiah dari Dosen Strategi Algoritma. Waktu kompetisi akan ditentukan kemudian oleh asisten.
12. Setiap anggota harus memahami proses pembuatan program, karena akan ada pertanyaan-pertanyaan yang harus dijawab per individu.
13. Program disimpan dalam folder `Tubes1_NIM` dengan NIM merupakan NIM anggota terkecil. Berikut merupakan struktur dari isi folder tersebut.
 - a. Folder **src** berisi **source code** (Folder `bot` kalian dengan struktur seperti folder "starter-bot")
 - b. Folder **bin** berisi **executable** (File `.jar`) dengan format **nama_kelompok.jar**
 - c. Folder **doc** berisi **laporan tugas besar** dengan format **nama_kelompok.pdf**
 - d. README untuk tata cara penggunaan yang minimal berisi:
 - i. Penjelasan singkat algoritma greedy yang diimplementasikan
 - ii. Requirement program dan instalasi tertentu bila ada
 - iii. Cara menggunakan program

iv. Author / identitas pembuat

Folder tersebut **di-zip** dengan format yang sama dengan nama folder. Link pengumpulan akan diberitahukan lebih lanjut oleh asisten.

14. Setiap kelompok harap mengisi nama kelompok dan anggotanya pada link <http://bit.ly/PendataanStima1>

Isi laporan:

- **Cover:** Cover laporan ada foto anggota kelompok (foto bertiga). Foto ini menggantikan logo “gajah” ganesha.
- **Bab 1:** Deskripsi tugas (dapat menyalin spesifikasi tugas ini).
- **Bab 2:** Landasan Teori.
 - Dasar teori (algoritma *greedy*) secara umum
 - Bagaimana pemanfaatan *game engine* (bagaimana menambahkan pemain, bagaimana menambahkan strategi *greedy* pada game engine, bagaimana menjalankan *game engine*, dll).
- **Bab 3:** Pemanfaatan strategi *greedy*.
 - Proses mapping persoalan Worms menjadi elemen-elemen algoritma Greedy (himpunan kandidat, himpunan solusi, fungsi seleksi, fungsi kelayakan, fungsi objektif)
 - Eksplorasi alternatif solusi *greedy* yang mungkin dipilih dalam persoalan Worms
 - Analisis efisiensi dari kumpulan alternatif solusi *greedy* yang dirumuskan
 - Analisis efektivitas dari kumpulan alternatif solusi *greedy* yang dirumuskan
 - Strategi *greedy* yang dipilih (yang akan diimplementasikan dalam program) beserta alasan dan pertimbangan pemilihan strategi tersebut.
- **Bab 4:** Implementasi dan pengujian.
 - Implementasi program dalam *game engine* yang digunakan (*pseudocode* yang cukup detail dengan komentar untuk pembaca kode agar mudah dipahami), misalnya menampilkan nama file program, dll.
 - Penjelasan struktur data yang digunakan dalam program Worms dan struktur data tambahan jika ada.
 - Analisis dari desain solusi algoritma *greedy* yang diimplementasikan pada setiap **pengujian** yang dilakukan. Misalnya adalah apakah strategi *greedy* berhasil mendapatkan nilai optimal, lalu jika tidak, dalam kondisi seperti apa strategi *greedy* tidak berhasil mendapatkan nilai optimal, dll.
- **Bab 5:** Kesimpulan dan saran.
- **Daftar Pustaka.**

Keterangan laporan:

1. Laporan ditulis dalam bahasa Indonesia yang baik dan benar.
2. Identitas per halaman harus jelas (misalnya : halaman, kode kuliah).

Penilaian:

1. Bagian 1: Desain Solusi Algoritma Greedy ditulis dalam Laporan (50%)

- a. Mapping persoalan Worms ke dalam elemen-elemen algoritma Greedy (5%)
- b. Eksplorasi alternatif-alternatif solusi persoalan Worms dengan algoritma Greedy (10%). Semakin banyak alternatif yang dieksplorasi maka akan semakin bagus nilainya.
- c. Analisis efisiensi secara teoritis dan alternatif solusi persoalan (5%)
- d. Analisis efektivitas secara teoritis dari alternatif solusi persoalan (5%)
- e. Analisis dari Desain solusi Algoritma Greedy yang diusulkan untuk diimplementasikan dalam program/coding (10%)
- f. Kualitas Strategi Algoritma yang diusulkan (15%)

2. Bagian 2: Implementasi Program / Coding (50%)

- a. Kesesuaian strategi Greedy yang dituliskan dengan implementasi dan saat demo (20%)
- b. Kreativitas program (10%)
- c. Demo pemahaman program (20%)

3. Bagian 3: Kompetisi dan Komponen Lainnya

- a. Beberapa kelompok pemenang (akan ditentukan jumlahnya nanti) pada saat kompetisi akan mendapatkan bonus nilai 5 (bonus 5%)
- b. Bonus dalam membuat video kelompok (bonus 10%)

--- Selamat Mengerjakan, it's not worth it if you're not have fun ---