

Petunjuk Pelaksanaan dan Spesifikasi Tugas Besar I if2032

Simulasi PokeRanch

Daftar Isi

[Daftar Isi](#)

[Pedoman Tugas Besar I if2032](#)

[Spesifikasi Tugas Besar I if2032](#)

[Pengantar](#)

[Gameplay](#)

[Area Luar](#)

[Kota](#)

[Mode Free Roam](#)

[Mode Battle](#)

[Objek](#)

[PEMAIN](#)

[AREA \(DUNIA LUAR / KOTA\)](#)

[MONSTER](#)

[SKILL MONSTER](#)

[ITEM](#)

[Bonus](#)

Petunjuk Pelaksanaan Tugas Besar I if2032

- 1 Tugas dikerjakan dalam kelompok beranggotakan 6 orang dari kelas yang sama (tidak boleh lintas kelas). Karena jumlah peserta tidak bulat dibagi 6, mungkin ada kelompok berjumlah 7 orang. Kelompok yang beranggotakan lebih dari 6 orang diwajibkan untuk mengerjakan bonus nomor 2 (umur), nomor 3 (mekanisme waktu siang/malam), dan tipe elemen ditambah menjadi 4.
- 2 Setiap kelompok akan dibimbing oleh 2 orang asisten, dan tiap-tiap kelompok wajib melakukan asistensi minimal 3 kali sesuai jadwal yang ditentukan.
- 3 Setiap kali asistensi, anggota kelompok dan asisten diharuskan mengisi form asistensi dengan lengkap. Setiap anggota kelompok sebisa mungkin hadir pada saat asistensi. Form asistensi hanya boleh diisi dan diparaf saat asistensi berlangsung (tidak boleh menyusul). Pada saat asistensi, kelompok menceritakan progress dan dipersilakan untuk bertanya kepada asisten jadi siapkan pertanyaan sebelum asistensi.
- 4 Tugas dibagi menjadi 3 tahap, dimana *deliverables* untuk setiap tahap adalah sebagai berikut:
 - a Tahap I
 - *Hardcopy* dokumen rancangan kelas
Rancangan kelas yang dikumpulkan adalah yang akan diimplementasikan pada tahap berikutnya. Jika nantinya implementasi program tidak konsisten dengan rancangan kelas, maka akan dikenakan penalti nilai. Isi dokumen mencakup daftar kelas beserta deskripsi masing-masing kelas (termasuk atribut dan method utamanya).
 - b Tahap II
 - Prototipe program yang telah mengimplementasikan sebagian besar fitur program final
 - Pengumpulan dilakukan di *uploader* yang akan disediakan di <http://milestone.if.itb.ac.id> dengan format nama file pengumpulan: TB1-2-XX-YY.zip (XX: 2 digit nomor kelas, YY: 2 digit nomor kelompok)
 - c Tahap III
 - Program versi final
 - Pengumpulan dilakukan di *uploader* yang akan disediakan di <http://milestone.if.itb.ac.id> dengan format nama file pengumpulan: TB1-XX-YY.zip (XX: 2 digit nomor kelas, YY: 2 digit nomor kelompok)
 - File program dengan format penamaan yang sama di-*burn* ke CD
 - *Hardcopy* laporan pengerjaan Tugas Besar I
 - Form penilaian Tugas Besar I
- 5 Untuk tugas program, Anda **tidak diperbolehkan** menggunakan *framework* apapun dan harus bisa di-*compile* di Linux. Jika menggunakan library luar, harus ditanyakan terlebih dahulu kepada asisten. Tampilan program tanpa GUI (pakai ASCII saja) dalam bahasa C++. Anda diperbolehkan menggunakan IDE (*Integrated Development Environment*)

apapun, misal Dev-Cpp, Visual C++, dan sebagainya asalkan bisa di-*compile* di Linux, jadi pastikan program Anda dites terlebih dahulu di Linux sebelum dikumpulkan.

6 Agenda Pelaksanaan Tugas Besar I IF2032

No.	Tanggal	Rincian Kegiatan
1.	4 Maret 2013	Rilis Tugas Besar I
2.	4 - 5 Maret 2013	Pendaftaran kelompok melalui http://milestone.if.itb.ac.id/ sampai pukul 12.00
3.	5 Maret 2013	Pengumuman daftar kelompok final beserta asisten pembimbingnya (pukul 17.00)
4.	6 – 8 Maret 2013	Waktu Asistensi I
5.	11 Maret 2013	Pengumpulan Tahap I <ul style="list-style-type: none"> ○ Di Lab. Programming paling lambat pukul 17.00
6.	11 – 15 Maret 2013	Waktu Asistensi II
7.	18 Maret 2013	Pengumpulan Tahap II <ul style="list-style-type: none"> ○ File program di-<i>upload</i> di http://milestone.if.itb.ac.id/ paling lambat pukul 17.00
8.	19 – 22 Maret 2013	Waktu Asistensi III
9.	25 Maret 2013	Pengumpulan Final <ul style="list-style-type: none"> ○ File program di-<i>upload</i> di http://milestone.if.itb.ac.id/ paling lambat pukul 17.00 ○ Laporan dan CD file program dikumpulkan di Lab. Programming paling lambat pukul 17.00

Spesifikasi Tugas Besar I if2032

Pengantar

Pada simulasi ini Anda akan memainkan tokoh bernama Okep yang hidup di kota Poke. Seperti kebanyakan remaja di kotanya, Okep gemar bermain dengan monster yang bisa diadu dengan monster lain. Hasil yang didapat dari pertarungan adalah *experience* (untuk monster) dan *money* (untuk Okep). Objektif dari permainan ini adalah mengumpulkan semua jenis monster dan melatih mereka agar menjadi lebih kuat dan bisa diadu di stadium.

Di luar aktivitas mengadu monster, Okep bisa berkeliling kota dan mengunjungi beberapa bangunan seperti Toko, Stadium, Combinatorium, dan Rumah Okep sendiri. Masing-masing bangunan memiliki fungsinya sendiri.

Terdapat beberapa jenis dan tipe elemen monster yang menentukan kemampuan sang monster. Untuk memperoleh jenis yang unggul, dua jenis monster yang berbeda dapat digabungkan menjadi satu monster dan menghasilkan kemampuan yang baru.

Anda diminta untuk membuat sebuah program tanpa GUI (Graphical User Interface) dalam bahasa C++ yang mampu mensimulasikan kehidupan Okep.

Gameplay

Pada awal PokeRanch dijalankan, Anda dapat memilih untuk memulai simulasi baru (*new game*) atau menggunakan simulasi lampau (*load game*) yang telah disimpan.

Untuk memulai simulasi baru digunakan perintah berikut:

```
new <nama-file>
```

command ini digunakan untuk membuat profil pemain baru dan dicatat dalam sebuah file eksternal.

Untuk menggunakan simulasi lampau yang telah disimpan digunakan perintah berikut:

```
load <nama-file>
```

command ini digunakan untuk melanjutkan permainan dengan profil yang sudah pernah dibuat sebelumnya.

Terdapat dua area utama yang dapat dikunjungi oleh pemain yaitu: area luar dan kota.

1 Area Luar

Area luar merupakan area di luar kota yang merupakan tempat yang tepat untuk menangkap monster si Okep. Area luar hanya bisa diakses dari *street* (jalan) yang ada di kota. Ketika berjalan di area luar, terdapat kemungkinan bagi Okep untuk bertemu dengan monster liar

secara *random*. Jika monster yang dimiliki Okep cukup kuat untuk melemahkan monster lawan, maka monster tersebut bisa ditangkap menggunakan 'Monster Ball' dan menjadi milik Okep.

2 Kota

Kota merupakan tempat tinggal Okep dan monsternya. Di kota terdapat beberapa tempat yang memungkinkan Okep beraktivitas dengan monster-monsternya. Tempat-tempat tersebut adalah:

- Home

Tempat ini merupakan tempat tinggal Okep dan monster - monsternya. Di tempat ini pemain dapat melakukan aksi: *sleep*, *save*, atau *load* pada lokasi / *grid* tertentu. Aksi (*sleep*) akan menyebabkan revitalisasi kondisi monster. Berikut adalah perintah yang digunakan untuk melakukan ketiga aksi tersebut:

<code>sleep</code>
<i>command</i> ini digunakan untuk me-revitalisasi tenaga dan energi monster. Apabila Anda mengimplementasi bonus 2 (umur monster), dan bonus 3 (siang-malam)
<code>save</code>
<i>command</i> ini digunakan untuk menyimpan <i>state</i> permainan seperti tanggal, jumlah uang, jumlah menang / kalah dalam pertandingan, status monster, dsb.
<code>load <nama-file></code>
<i>command</i> ini digunakan untuk mengembalikan <i>state</i> permainan ke <i>state</i> terakhir yang disimpan.

- Store

Tempat ini digunakan Okep untuk menjual atau membeli *item* yang berguna dalam pertarungan monster. Daftar item yang bisa dibeli/dijual akan dijelaskan pada bagian selanjutnya. Berikut adalah perintah yang digunakan untuk melakukan kedua aksi tersebut :

<code>sell <nama-item> [jumlah-item]</code>
Digunakan untuk menjual item tertentu. Harga jual ditentukan sendiri oleh peserta.
<code>buy <nama-item> [jumlah-item]</code>
Digunakan untuk membeli item tertentu. Harga beli ditentukan sendiri oleh peserta.
<i>Catatan:</i> argumen dalam kurung siku bersifat opsional. Apabila tidak digunakan, maka diasumsikan bernilai satu.

- Combinatorium

Tempat ini merupakan laboratorium khusus yang digunakan untuk mengombinasikan berbagai jenis monster untuk menghasilkan keturunan dengan karakteristik yang baru. Berikut adalah perintah yang digunakan untuk menggabungkan kedua jenis monster:

```
combine <monster-1> <monster-2>
```

Command ini digunakan untuk menggabungkan antara monster pertama dengan monster kedua yang diinginkan. Isi parameter <monster> dibebaskan pada Anda, bisa menggunakan nama monster atau id-monster.

- Stadium

Tempat ini merupakan tempat untuk bertarung dan bertaruh dengan para pemilik monster lainnya di kota. Berikut adalah perintah yang digunakan untuk bertarung dan bertaruh:

```
battle <bet>
```

Command ini digunakan untuk memasang besarnya nilai uang taruhan dan memulai 'mode battle' (akan dijelaskan pada bagian selanjutnya).

- Street

Tempat ini merupakan penghubung antara tempat - tempat yang lain baik yang ada di kota maupun di area luar. Setiap kali Okep ingin pindah ke suatu tempat harus melalui *street* terlebih dahulu.

Dalam simulasi ini terdapat dua mode utama yaitu:

1 Mode Free Roam

Pada mode ini, Anda dapat membawa Okep berkeliling kota dan area luar serta melakukan berbagai hal di tempat-tempat yang ada. Terdapat beberapa aksi yang dapat dilakukan Okep pada mode Free Roam yaitu :

- Teleport, digunakan untuk pindah lokasi pada grid tertentu di peta. Perintah yang digunakan adalah :

```
teleport <nama-tempat>
```

Misalkan; apabila ingin pindah dari rumah ke jalan, Okep harus berada pada grid yang terletak di depan pintu rumah baru bisa menggunakan *command* ini. Kemudian setelah *command* dijalankan, Okep akan muncul pada grid yang terletak di depan bangunan rumah pada peta.

- Move, digunakan untuk berpindah ke kiri, kanan, atas, atau bawah. Perintah yang digunakan adalah :

```
move <arah> [jumlah-langkah]
```

Apabila langkah pemain mencapai atau melebihi batas peta, pemain akan berhenti pada ujung batas peta.

- List-monster, digunakan untuk menampilkan daftar monster yang dimiliki Okep. Perintah yang digunakan adalah:

```
list-monster
```

Selain menampilkan daftar monster, *command* ini bisa digunakan untuk menampilkan status monster. Desain tampilan dan data yang ditampilkan diserahkan pada Anda.

- List-item, digunakan untuk menampilkan daftar item yang dimiliki Okep. Perintah yang digunakan adalah :

```
list-item
```

Selain menampilkan daftar item, *command* ini bisa digunakan untuk menampilkan deskripsi item (kegunaan, efek pada status, dsb). Desain tampilan dan data yang ditampilkan diserahkan pada Anda.

- Status, digunakan untuk menampilkan uang, jumlah menang, dan jumlah kalah. Perintah yang digunakan adalah :

```
Status
```

Selain data yang sudah disebutkan di atas, Anda bisa berkreasi dengan menampilkan data lain seperti jumlah dan jenis monster yang pernah dipelihara, dsb.

- Set, digunakan untuk membuat monster tertentu menjadi monster yang digunakan. Perintah yang digunakan adalah :

```
set <nama-monster> atau  
set <nomor-monster>
```

Anda dibebaskan untuk menentukan parameter mana yang akan digunakan.

- Dismiss, digunakan untuk melepaskan monster ke alam liar. Perintah yang digunakan adalah:

```
dismiss <nama-monster>  
atau  
dismiss <nomor-monster>
```

Anda dibebaskan untuk menentukan parameter mana yang akan digunakan.

- Help, digunakan untuk menampilkan aksi-aksi yang ada beserta kegunaannya. Perintah yang digunakan adalah :

```
help
```

aksi-aksi yang ditampilkan adalah *command-command* yang sudah dijelaskan sebelumnya.

2 Mode Battle

Pada mode ini, Okep akan menggunakan monsternya untuk bertarung dengan monster lain. Mode pertarungan bisa dimulai ketika pemain berinteraksi dengan lawan di stadium (command bet <money>) atau ketika berkelana di area luar kota. Pada layar akan tampil status dari monster pemain dan monster lawan. Terdapat beberapa aksi yang dapat dilakukan Okep pada mode Battle yaitu:

- Run, digunakan untuk melarikan diri dan menghindari pertarungan. Perintah yang digunakan adalah:


```
escape
```

Command ini hanya bisa digunakan saat Okep bertarung dengan monster di area luar kota, sehingga otomatis command ini tidak memiliki efek ketika bertarung di stadium. Anda dibebaskan untuk menentukan besar peluang Okep melarikan diri dari pertarungan.

- Item, digunakan untuk menggunakan item yang ada. Perintah yang digunakan adalah :

```
item <nama-item> [item-argument]
```

Nama, jenis, dan efek item yang bisa digunakan dalam pertarungan dibebaskan pada kreativitas Anda. Namun Anda wajib membuat item yang diminta pada bagian 'ITEM' (akan dijelaskan pada bagian berikutnya).

Catatan: argumen dalam kurung siku bersifat opsional. Misal Okep menggunakan Potion pada monsternya sendiri, maka command yang perlu dieksekusi adalah:

```
item Potion Self
```

- Change, digunakan untuk mengganti monster pemain. Perintah yang digunakan adalah:

```
change <nama-monster>  
atau  
change <nomor-monster>
```

- Skill, digunakan untuk menggunakan skill dari monster yang sedang bertarung. Perintah yang digunakan adalah:

```
skill <nama-skill>  
atau  
skill <nomor-skill>
```

Nama, jenis, dan efek skill yang bisa digunakan oleh monster dibebaskan pada kreativitas Anda. Namun Anda wajib mengikuti petunjuk skill yang diminta pada bagian 'SKILL MONSTER' (akan dijelaskan pada bagian berikutnya).

Objek

Dalam permainan ini, terdapat beberapa objek-objek yang perlu diperhatikan dan wajib diimplementasi.

PEMAIN

Dalam permainan, karakter atau pemain akan memiliki beberapa atribut yang menentukan status dari pemain sekarang. Atribut minimal yang harus diimplementasikan pada pemain adalah:

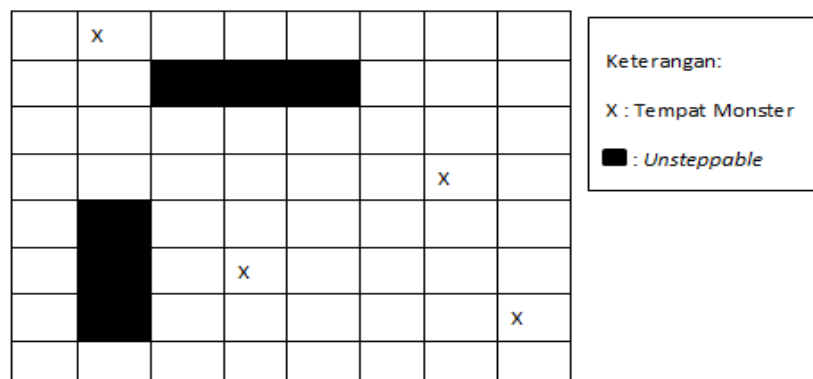
- 1 **Nama** – perlu diisikan pada saat pemain memulai permainan (default: **Okep**)
- 2 **Uang** – merupakan jumlah uang yang sedang dimiliki oleh pemain (satuan uang boleh ditentukan sendiri, misal: Rp, \$, Zeny, dan lain-lain).
- 3 **Backpack** – merupakan tas yang selalu dibawa oleh pemain dengan isi minimal sebagai berikut:
 - a **List-Item** – berisikan item-item yang sedang dimiliki oleh pemain (kapasitas boleh ditentukan atau tidak)
 - b **List-Monster** – berisikan monster-monster yang telah didapatkan oleh pemain. (kapasitas boleh ditentukan atau tidak)

AREA LUAR

Selain pemain, terdapat objek lain, yakni area atau peta untuk pemain berjalan-jalan mencari monster untuk bertarung atau ditangkap. Secara umum, area akan terdiri atas sekumpulan koordinat yang memiliki status tertentu (misal: *stepable*, *unstepable*, dan sebagainya). Dalam hal ini, area juga memerlukan koordinat **random** yang apabila pemain berpijak pada koordinat tersebut, pemain akan melawan monster pada koordinat tersebut. Implementasi **random** ini dibangkitkan pada saat:

- 1 Pemain memasuki area tersebut, dan
- 2 Pemain baru selesai melawan monster pada area tersebut.

Contoh dari area akan diimplementasikan seperti gambar berikut:



Selain seperti deskripsi diatas, area juga dapat berupa kota. Lebih jelasnya dapat mengacu ke Bagian Gameplay - Kota.

MONSTER

Yang tidak kalah penting dari permainan ini adalah berbagai jenis monster yang memiliki karakteristik masing-masing. Dalam hal ini, tiap monster perlu memiliki atribut minimal sebagai berikut:

- 1 **Nama** – pengenalan unik monster tersebut, bisa diisikan oleh pemain.
- 2 **Level** – tingkat level monster saat ini. Dalam sistem permainan ini, dikenal istilah evolusi untuk monster. Tiap monster akan berevolusi (berubah jenis spesies) pada level tertentu. Misalnya: **Pikachu** pada level 20 akan menjadi **Raichu**.
- 3 **Experience** – merupakan angka pengalaman monster yang bertambah seiring bertambahnya jumlah monster yang dikalahkan.
- 4 **Jenis Spesies** – merupakan pengenalan jenis dari monster ini. Jenis spesies akan berubah setelah monster mencapai level tertentu. (minimal terdapat 6 jenis spesies untuk tiap elemen).
- 5 **Tipe Elemen** – merupakan tipe monster yang akan berpengaruh pada serangan monster lawan (misal: serangan tipe air akan superior terhadap monster tipe api). Minimal jumlah tipe elemen adalah 3 tipe.
- 6 **Status** – merupakan keadaan monster saat ini. Dalam hal ini status dapat terdiri atas *Health Point*, *Mana Point*, dan sebagainya.
- 7 **Bonus Uang** – jumlah ini akan didapatkan oleh pemain setelah pemain berhasil mengalahkan monster ini (**bukan menangkap**).
- 8 **Bonus Experience** – jumlah ini akan didapatkan untuk menambah *experience* monster milik pemain setelah mengalahkan monster lawan.

SKILL MONSTER

Dalam bertarung, monster menggunakan keahlian (*skill*) tertentu. Pada permainan ini, beberapa atribut yang perlu dimiliki oleh *skill* monster adalah:

- 1 **Jenis serangan** – nama dari *skill*. Tiap jenis serangan memiliki level minimum monster untuk mempelajarinya.
- 2 **Tipe Elemen** – merupakan tipe serangan (misal: api, air) yang akan berkorelasi dengan tipe monster lawan.
- 3 **Damage** – merupakan besaran serangan dari *skill*.
- 4 **Efek** – merupakan efek yang ditimbulkan dari serangan (misal: *healing*, *burning*, *poisoned*, *confused*, *paralyzed*, dan sebagainya).
- 5 **Cost** – biaya yang diperlukan untuk menggunakan *skill*. Hal ini akan berkorelasi dengan *Mana Poin / PP*, *Health Point*, dan lain-lain.

ITEM

Untuk membantu pemain dalam bertarung melawan monster, menangkap monster, dan sebagainya, tersedia item-item tertentu dalam permainan. Pada permainan ini, terdapat item yang wajib ada:

- 1 **Potion** – item yang digunakan untuk menyembuhkan *Health Point* atau *Mana Point* dari monster.
- 2 **Monster Ball** – item yang digunakan untuk menangkap monster yang sudah “sekarat” dalam pertarungan.
- 3 **Stat Permanent Increase** – variasi item yang digunakan untuk menambah status perang dari monster, misal menambah maksimum *Health Point*, dan sebagainya.
- 4 **Monster Egg** – item yang dibeli oleh pemain untuk mendapatkan monster tanpa harus menangkapnya.

Bonus

Terdapat beberapa bonus dalam pengerjaan tugas besar ini, yakni sebagai berikut:

- 1 **Shortcut command** – terdapat cara singkat untuk memberi komando, misalnya: T untuk teleport, dan sebagainya.
- 2 **Monster memiliki umur** – terdapat atribut tambahan pada monster yakni umur. Mekanisme pertambahan umur dan peruntukkan umur ini dibebaskan. (misalnya: monster akan mati pada umur tertentu).
- 3 **Mekanisme waktu siang dan malam** – terdapat pewaktuan malam dan siang yang berpengaruh pada penambahan umur monster. Selain itu, waktu siang dan malam juga akan berpengaruh pada monster yang muncul pada area tertentu. (misalnya: pada waktu malam, monster yang timbul hanya burung hantu, serigala, dan lain-lain).
- 4 **Warna** – tampilan ASCII warna-warni.
- 5 **Multi-player** – terdapat mekanisme *multiplayer turn-based* dengan dua saved file berbeda dimainkan dalam satu *command prompt* bergantian dalam mode bertarung.
- 6 **Analisis program menggunakan Valgrind (Memcheck)** – menganalisis *memory* yang digunakan oleh program, berapa besar *memory* yang digunakan dan bagaimana kondisi *memory leak* yang terjadi (jika ada). Jadikan bab lampiran di laporan akhir.