IF 1210 - Dasar Pemrograman LAPORAN TUGAS BESAR



oleh:

Kelas 07 - Kelompok C

165232007 - Mikael Adrian Widagdo 16523137 - Muhammad Jafar Fadli 16523217 - Faiz Arkan Waskitazaman 19623217 - Nawaf Amjad Rizqi Aldaha Ismail 19623227 - Jessica Allen

Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung Tahun Ajaran 2023/2024

Pernyataan Kelompok

165232007 - Mikael Adrian Widagdo 16523137 - Muhammad Jafar Fadli 16523217 - Faiz Arkan Waskitazaman 19623217 - Nawaf Amjad Rizqi Aldaha Ismail 19623227 - Jessica Allen

Saya menyatakan bahwa saya mengerjakan tugas besar ini dengan sejujur-jujurnya, tanpa menggunakan cara yang tidak dibenarkan. Apabila di kemudian hari diketahui saya mengerjakan tugas besar ini dengan cara yang tidak jujur, saya bersedia mendapatkan konsekuensinya, yaitu mendapatkan nilai E pada mata kuliah IF1210 Dasar Pemrograman Semester 2 2023/2024.

Daftar Isi

Pernyataan Kelompok	1
Daftar Isi	2
Daftar Tabel	3
Daftar Gambar	4
Deskripsi Persoalan.	6
Daftar Pembagian Kerja Anggota Kelompok	7
Checklist Hasil Rancangan, Implementasi, dan Testing Setiap Primitif	9
Desain Perintah	10
Desain Kamus Data	14
Desain Dekomposisi Algoritmik dan Fungsional Program	20
Spesifikasi Program	28
Hasil Pengujian Program.	92
Lampiran	110

Daftar Tabel

Tabel 1 Pembagian Kerja Anggota Kelompok	8
Tabel 2 Checklist	9
Tabel 3 Desain Kamus Data	19
Tabel 4 Notasi Algoritma	91
Tabel 5 Hasil Pengujian Program	109

Daftar Gambar

Gambar 9.1 Welcome Page	92
Gambar 9.2 Command Not Found	92
Gambar 9.3 Register	92
Gambar 9.4 Login - Username tidak terdaftar	93
Gambar 9.5 Login - Password salah	93
Gambar 9.6 Login - Admin.	93
Gambar 9.7 Logout - Belum Login	93
Gambar 9.8 Logout - Berhasil	93
Gambar 9.9 Menu Help - Sebelum Login	94
Gambar 9.10 Menu Help - Admin	94
Gambar 9.11 Menu Help - Setelah Login	95
Gambar 9.12 Monster - OWCA-DEX.	95
Gambar 9.13 Potion - Detail Potion.	95
Gambar 9.13.1 Potion - Detail Potion dalam Pertarungan	95
Gambar 9.14 Inventory	96
Gambar 9.15 Inventory - Detail Inventory	96
Gambar 9.16 Battle	97
Gambar 9.17 Battle - Turn.	98
Gambar 9.18 Battle - Results	
Gambar 9.19 Arena	99
Gambar 9.20 Arena - Turn.	99
Gambar 9.21 Arena - Potion	99
Gambar 9.22 Arena - Results - Potion	100
Gambar 9.23 Arena - Results - Attack	
Gambar 9.24 Arena - Quit	100
Gambar 9.25 Shop - Welcome	101
Gambar 9.26 Shop - Lihat	101
Gambar 9.27 Shop - Beli Gagal	101
Gambar 9.28 Shop - Beli Berhasil	
Gambar 9.29 Shop - Keluar	
Gambar 9.30 Lab - Welcome	103
Gambar 9.31 Lab - Upgrade	103
Gambar 9.32 Lab - Invalid Input	
Gambar 9.33 Shop Management - Welcome.	104

Gambar 9.34 Shop Management - Lihat	104
Gambar 9.35 Shop Management - Tambah	104
Gambar 9.36 Shop Management - Ubah	105
Gambar 9.37 Shop Management - Hapus	105
Gambar 9.38 Shop Management - Keluar	
Gambar 9.39 Monster Management - Welcome	106
Gambar 9.40 Monster Management - Tampil	106
Gambar 9.41 Monster Management - Tambah	106
Gambar 9.42 Load - Error	106
Gambar 9.43 Save - Existing File.	107
Gambar 9.44 Save - File Baru	107
Gambar 9.45 Exit	107
Gambar 9.46 Exit - Save.	107
Gambar 9.47 Monster Ball - Gagal	107
Gambar 9.48 Monster Ball - Berhasil	108
Gambar 9.49 Jackpot - Welcome	108
Gambar 9.50 Jackpot - Result	108
Gambar 9.51 Peta Kota Danville - Titik Awal	109
Gambar 9.52 Peta Kota Danville - Obstacle	109

Deskripsi Persoalan

Pada tugas besar mata kuliah IF1210 - Dasar Pemrograman ini, peserta mata kuliah ditugaskan untuk membuat sebuah game dimana pengguna adalah seorang agen yang bertugas membantu Agent P untuk mengalahkan Dr. Asep Spakbor. Ia telah menciptakan monster-monster mengerikan yang mengancam keamanan kota Danville. Untuk mengalahkannya, agen-agen (pengguna) harus mengalahkan monster-monster kuat dengan strategi yang matang pada sebuah hutan terpencil.

Game ini terdiri atas beberapa fitur, yaitu fitur register, login, logout, menu & help, monster, potion, inventory, battle, arena, shop & currency, laboratory, shop management, monster management, load, save, dan exit. Beberapa fitur dalam program ini hanya dapat diakses oleh admin, yaitu shop management dan monster management.

Pada dasarnya, seluruh perintah pada game ini berdasar pada manipulasi array yang dilakukan untuk mengakses data-data yang diperlukan untuk menjalankan game. Maka, permasalahan awal yang perlu diselesaikan adalah cara untuk mengubah data csv ke dalam bentuk array yang dapat digunakan kembali dalam fungsi-fungsi atau prosedur yang diperlukan dalam game. Setelah itu, data tersebut dapat digunakan untuk merealisasikan fitur-fitur yang diperlukan dalam game. Data hasil perubahan juga perlu diubah kembali dalam bentuk csv untuk penyimpanan data pengguna.

Daftar Pembagian Kerja Anggota Kelompok

Fitur	Implementasi	NIM Desainer	NIM Coder	NIM Tester
F00 - Random Number Generator	Fungsi lcg_x	16523137	16523137 19623227	Semua Anggota
F01 - Register	Prosedur register	16523137 19623227	16523137 19623227	Semua Anggota
F02 - Login	Prosedur login	16523137	16523137 19623227	Semua Anggota
F03 - Logout	Prosedur logout	16523137	16523137 19623227 19623217	Semua Anggota
F04 - Menu Help	Prosedur help	16523137 19623227	16523137 19623227	Semua Anggota
F05 - Monster	Fungsi monster Fungsi read_monster Fungsi update_monster	16523137	16523137 19623227	Semua Anggota
F06 - Potion	Fungsi read_potion Fungsi update_potion	16523137	16523137 19623227	Semua Anggota
F07 - Inventory	Prosedur inventory	16523137 19623227	16523137 19623227	Semua Anggota
F08 - Battle	Prosedur battle	16523137	16523137 19623227	Semua Anggota
F09 - Arena	Prosedur arena	16523137	16523137 19623227 16523217	Semua Anggota
F10 - Shop Currency	Fungsi update_oc Prosedur shop	16523137	16523137 19623227	Semua Anggota
F11 - Laboratory	Prosedur	16523137	16523137	Semua Anggota

	laboratory	19623227	19623227	
F12 - Shop Management	Prosedur shop_manageme nt	16523137	16523137 19623227	Semua Anggota
F13 - Monster Management	Prosedur monster_manage ment	16523137	16523137 19623227	Semua Anggota
F14 - Load	Prosedur load	16523137	16523137 19623227	Semua Anggota
F15 - Save	Prosedur save	16523137	16523137 19623227	Semua Anggota
F16 - Exit	Prosedur exit	16523137	16523137 19623227	Semua Anggota
B03 - Monster Ball	Item monster ball	16523137	16523137 19623227	Semua Anggota
B04 - Jackpot	Prosedur jackpot	16523137	16523137 19623227	Semua Anggota
B05 - Peta Kota Danville	Peta Danville dalam prosedur main	16523137 19623227	16523137 19623227	Semua Anggota

Tabel 1 Pembagian Kerja Anggota Kelompok

Laporan:

- 16523137: Desain Kamus Data, Daftar Pembagian Kerja Kelompok, Checklist, Referensi
- 16523217: Flowchart
- 19623227: Desain Perintah, Notasi Algoritma, Hasil Pengujian Program, Deskripsi Persoalan

Checklist Hasil Rancangan, Implementasi, dan Testing Setiap Primitif

Fitur	Design	Implementasi	Testing
F00 - Random Number Generator	✓	✓	✓
F01 - Register	✓	✓	✓
F02 - Login	✓	✓	✓
F03 - Logout	✓	1	✓
F04 - Menu Help	✓	1	✓
F05 - Monster	✓	1	✓
F06 - Potion	✓	✓	✓
F07 - Inventory	✓	1	✓
F08 - Battle	✓	✓	✓
F09 - Arena	✓	✓	✓
F10 - Shop Currency	✓	✓	✓
F11 - Laboratory	✓	✓	✓
F12 - Shop Management	✓	✓	✓
F13 - Monster Management	✓	1	✓
F14 - Load	✓	✓	✓
F15 - Save	✓	✓	✓
F16 - Exit	✓	1	✓
B03 - Monster Ball	✓	1	✓
B04 - Jackpot	✓	1	1
B05 - Peta Kota Danville	✓	✓	1

Tabel 2 Checklist

Desain Perintah

1. Register

<u>input</u>

username → pengguna akan diminta untuk memasukkan username password → pengguna akan diminta untuk memasukkan password monster → pengguna akan diminta untuk memilih monster yang diinginkan

output

Prosedur akan *assign* id kepada user, memeriksa ketersediaan username, serta menyimpan data user, kemudian memunculkan pesan berhasil pada layar.

2. Login

input

username \rightarrow pengguna akan diminta untuk memasukkan username password \rightarrow pengguna akan diminta untuk memasukkan password

<u>output</u>

Prosedur akan memeriksa kesesuaian username dan password dengan yang ada pada data, kemudian memunculkan pesan atas langkah selanjutnya yang perlu diambil setelah berhasil login. Jika password dan username tidak sesuai, prosedur akan memunculkan pesan gagal.

3. Logout

input

(tidak ada)

output

Program akan keluar dari akun pengguna jika pengguna sudah login sebelumnya. Sebaliknya, jika pengguna belum login, prosedur ini akan memunculkan pesan gagal.

4. Menu Help

input

(tidak ada)

output

Memunculkan *list of commands* (daftar perintah) yang dapat digunakan oleh pengguna dalam game.

5. Monster

Terdapat prosedur untuk menampilkan list monster yang dimiliki pengguna, fungsi untuk membaca level monster, serta prosedur untuk meningkatkan level monster.

6. Potion

<u>input</u>

(tidak ada)

output

Terdapat fungsi untuk membaca kuantitas potion atau item pengguna serta prosedur untuk meng-*update* kuantitas potion atau item untuk pengguna.

7. Inventory

<u>input</u>

nomor id → pengguna akan diminta untuk memasukkan nomor id yang spesifikasinya ingin ditampilkan

output

Menampilkan list item yang dimiliki pengguna beserta spesifikasinya bila diminta oleh pengguna.

8. Battle

<u>input</u>

monster → pengguna akan diminta untuk memilih monster yang ingin digunakan untuk bertarung

 $perintah \rightarrow pengguna \ akan \ diminta \ untuk \ memilih \ perintah \ yang \ ingin \ dijalankan$

output

Memunculkan hasil dari setiap pertarungan yang terjadi serta kondisi akhir pengguna.

9. Arena

input

perintah → pengguna akan diminta untuk memasukkan perintah yang ingin dilakukan ketika berada di arena

<u>output</u>

Program akan memunculkan hasil pertarungan yang terjadi pada setiap turn dan setiap stage, kemudian memunculkan kondisi akhir pengguna

10. Shop & Currency

input

aksi \rightarrow pengguna akan diminta untuk memilih opsi aksi yang ingin dilakukan output

Program akan memunculkan pesan mengenai hasil proses pembelian serta memunculkan pesan error jika transaksi tidak dapat terjadi.

11. Laboratory

<u>input</u>

pilihan → pengguna akan diminta untuk memilih monster yang ingin di-upgrade konfirmasi → pengguna akan diminta untuk memastikan kembali upgrade yang ingin dilakukan

<u>output</u>

Bila O.W.C.A Coin mencukupi, program akan memunculkan pesan transaksi dan upgrade berhasil. Program juga akan memastikan kembali kepada pengguna atas upgrade yang akan terjadi dengan memunculkan pesan konfirmasi. Jika tidak, program akan memunculkan pesan error.

12. Shop Management (traversal admin)

input

pilihan → admin akan diminta untuk memilih aksi yang ingin dilakukan serta monster yang ingin dipilih

ubah → admin akan diminta untuk memasukkan perubahan yang ingin dilakukan dalam shop

<u>output</u>

Program dapat memunculkan list stock dari shop kepada admin, serta dapat menunjukkan pesan konfirmasi dan hasil perubahan jumlah stok dalam shop.

13. Monster Management

<u>input</u>

aksi → admin akan diminta untuk memilih aksi yang ingin dilakukan dalam Monster Management

opsi → admin akan diminta untuk memilih monster yang ingin dipilih

ubah \rightarrow admin akan diminta untuk memasukkan jenis perubahan dan jumlah yang ingin dilakukan kepada monster yang telah dipilih

konfirmasi → admin akan diminta untuk mengkonfirmasi pilihan perubahan yang ingin dilakukan

<u>output</u>

Program akan memunculkan pesan error jika perubahan gagal dilakukan dan pesan berhasil jika perubahan berhasil dilakukan.

14. Load

<u>input</u>

Memeriksa keberadaan folder dan mengambil data dari folder tersebut.

output

Memberikan pesan error jika folder tidak dapat diakses. Mengakses dan mengambil data yang diperlukan dari folder yang berhasil diakses.

15. Save

<u>input</u>

Data baru hasil perubahan dari pengguna.

<u>output</u>

Memunculkan pesan berhasil ketika penyimpanan berhasil dilakukan.

16. Exit

<u>input</u>

penyimpanan → pengguna akan diberikan opsi untuk melakukan penyimpanan pada file yang sudah diubah jika belum disimpan

output

Memunculkan pesan error jika input tidak valid. Jika input valid, program akan keluar.

17. Jackpot

<u>input</u>

konfirmasi → pengguna akan diminta untuk mengkonfirmasi akan mengikuti permainan

<u>output</u>

Memunculkan pesan error jika Coin yang dimiliki pengguna tidak cukup untuk bermain jackpot atau input tidak valid. Memunculkan hasil jackpot yang didapatkan oleh pengguna.

18. Utility

input

string \rightarrow menerima string untuk dipisah dan diperiksa sebagai integer non-negatif file \rightarrow menerima file csv untuk dilakukan perubahan

<u>output</u>

Program akan mengembalikan array dan boolean hasil pengecekan dan perubahan.

19. Main

<u>input</u>

 $\operatorname{command} \to \operatorname{pengguna}$ akan diminta untuk memasukkan perintah yang ingin dilakukan pada game

output

Program akan memunculkan pesan error jika command tidak dapat dilakukan, serta akan menunjukkan posisi agent dalam peta.

Desain Kamus Data

```
DEKLARASI TIPE DATA
    {Constant.py}
    type RowDataUser : < id : string,</pre>
                             username : string,
                             password : string,
                             role : string,
                             oc : string >
                            < id : string,
    type RowDataMonster:
                             type : string,
                             atk power : string,
                             def power : string,
                             hp : string >
    type RowDataMonsterInventory : < user id : string,</pre>
                                         monster id : string,
                                         level : string >
    type RowDataItemInventory : < user id : string,</pre>
                                     type : string,
                                     quantity : string >
    type RowDataMonstershop : < monster id : string,</pre>
                                 stock : string,
                                 price : string >
    type RowDataItemshop : < id : string,</pre>
                                 type : string,
                                 stock : string,
                                 price : string >
    type RowDataItem : < id : string,</pre>
                             type : string >
    data user : array of RowDataUser
    data monster : array of RowDataMonster
    data monster inventory : array of RowDataMonsterInventory
    data item shop : array of RowDataItemInventory
    data item inventory : array of RowDataMonstershop
    data monster shop : array of RowDataMonstershop
    data login : RowDataUser
    data item : array of RowDataItem
    monster level5 : string
```

```
monster : string
   jackpot header : string
   lab header : string
   shop header : string
   shop management header : string
   main welcome : string
   help header : string
   inventory header : string
   owcadex header : string
   monster header : string
    {Utility.py}
    function split str(string : string, delimiter : string) ->
array of string
   KAMUS LOKAL
       array: list of string
       limit : integer
   function read csv(file name : string) -> array of array of
string
   KAMUS LOKAL
       data: array of array of string
       row: array of string
    function write csv(file name : string, data : array of
array of string) -> None
   KAMUS LOKAL
       row: array of string
   function check int(string : string) -> boolean
    {F00 - Random Number Generator.py}
   function lcg x(n : integer) -> integer
    KAMUS LOKAL
       a : integer
       c : integer
       m : integer
    {F01 - Register.py}
   procedure register() -> None
    KAMUS LOKAL
       user id : string
       username : string
       password : string
        success : boolean
        list monster : array of string
```

```
monster id : string
    {F02 - Login.py}
    procedure login() -> None
    KAMUS LOKAL
       username : string
       password : string
        success : boolean
        exists : boolean
    {F03 - Logout.py}
   procedure logout() -> None
   {F04 - Menu Help.py}
   procedure help() -> None
   {F05 - Monster.py}
    function monster() -> None
    KAMUS LOKAL
        index monster : integer
        list monster : array of string
    function read monster(id : string, monster id : string) ->
string
    function update monster(id : string, monster id : string,
change lvl : integer) -> None
    {F06 - Potion.py}
    function read potion(id : string, type potion : string) ->
string
    update potion(id : string, type potion : string, change qty
: integer) -> None
    {F07 - Inventory.py}
   procedure inventory() -> None
    KAMUS LOKAL
        list inventory: array of array of string
       monster id : integer
       num : integer
        id : string
    {F08 - Battle.py}
    procedure battle() -> None
    KAMUS LOKAL
```

```
data monster agent : array of array of string
    data monster musuh : array of array of string
    list monster: array of string
    id musuh : integer
    monster id : integer
    stats musuh : array of string
    stats agent : array of string
    lvl agent : string
    hp terakhir : string
    attack result : float
    strength potion usage : boolean
    resilience potion usage : boolean
    healing potion usage : boolean
    kabur : boolean
    dmg multiplier : integer
    item exists : boolean
    turn : integer
    reward : integer
{F09 - Arena.py}
procedure arena() -> None
KAMUS LOKAL
   reward : array of integer
    total hadiah : integer
    total win : integer
    total dmg diberikan : float
    total dmg diterima : float
    data monster agent : array of array of string
    data monster musuh : array of array of string
    list monster : array of string
    stage : integer
    id musuh : integer
   monster id : integer
    stats musuh : array of string
    stats agent : array of string
    lvl agent : string
    hp terakhir : string
    attack result : float
    strength potion usage : boolean
    resilience potion usage : boolean
    healing potion usage : boolean
    kabur : boolean
    dmg multiplier : integer
    turn : integer
{F10 - Shop Currency.py}
```

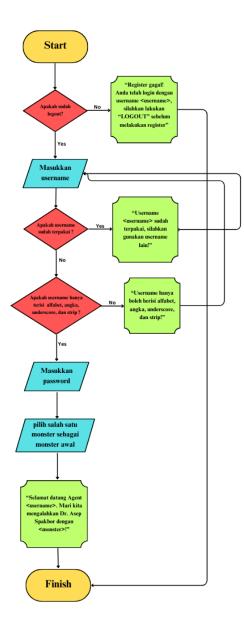
```
procedure shop() -> None
KAMUS LOKAL
    aksi : string
    lihat : string
   beli : string
    id : string
    item id : string
   monster id : string
    jumlah : string
{F11 - Laboratory.py}
procedure laboratory() -> None
KAMUS LOKAL
    list harga : array of integer
    list monster : array of string
    id : string
    lanjutkan : string
{F12 - Monster Management.py}
procedure monster management() -> None
KAMUS LOKAL
    aksi : string
    success : boolean
   nama : string
    attack : string
    defense : string
   hp : string
    tambahkan : string
{F13 - Shop Management.py}
procedure shop management() -> None
KAMUS LOKAL
    aksi : string
    lihat : string
    tambah : string
    ubah : string
   hapus : string
    id : string
   exists : boolean
   monster id : string
    item id : string
    stok baru : string
    harqa baru : string
    yakin : string
{F14 - Load.py}
```

```
procedure load() -> boolean
KAMUS LOKAL
    exists : boolean
    folder name : string
    argparser : ArgumentParser
    args : Namespace
{F15 - Save.py}
procedure save() -> None
KAMUS LOKAL
    folder name : string
    folder baru : boolean
{F16 - Exit.py}
procedure exit(saved : boolean) -> None
KAMUS LOKAL
    simpan : str
{B04 - Jackpot.py}
procedure jackpot() -> None
KAMUS LOKAL
    list_item : array of string
    list value : array of integer
    list monster : array of integer
    item1 : integer
    item2 : integer
    item3 : integer
    reward : integer
{main.py}
saved : boolean
command : string
danville : array of array of string
run game : boolean
x : integer
y : integer
```

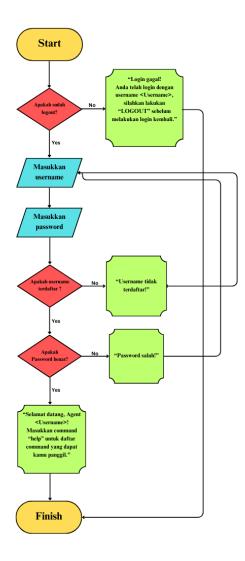
Tabel 3 Desain Kamus Data

Desain Dekomposisi Algoritmik dan Fungsional Program

F01 - Register

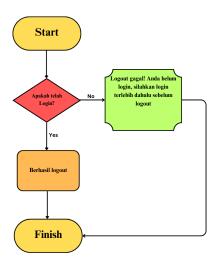


F02 - Login



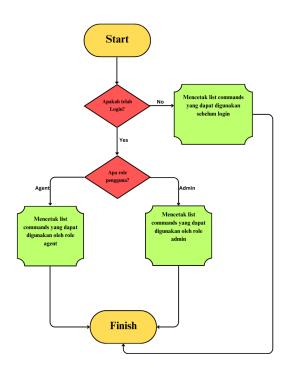
Gambar 7.2 Flowchart Login

F03 - Logout



Gambar 7.3 Flowchart Logout

F04 - Menu & Help



Gambar 7.4 Flowchart Menu dan Help

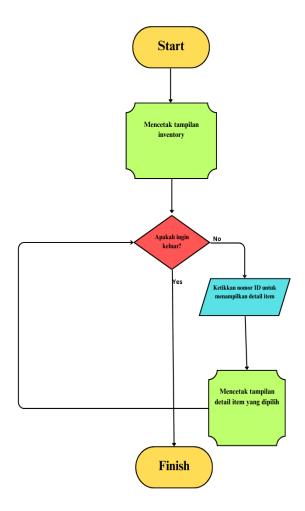
F05 - Monster

- monster = fungsi menampilkan OWCA DEX yang dimiliki user
- read_monster = fungsi membaca level monster untuk user
- update monster = fungsi update level monster untuk user

F06 - Potion

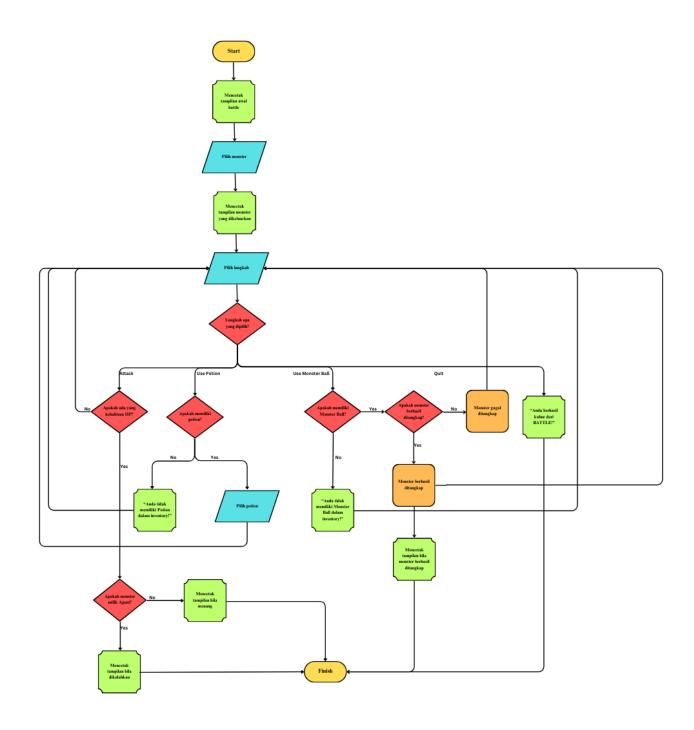
- read_monster = fungsi membaca kuantitas potion/item untuk user
- update_potion = fungsi update kuantitas potion/item untuk user

F07 - Inventory

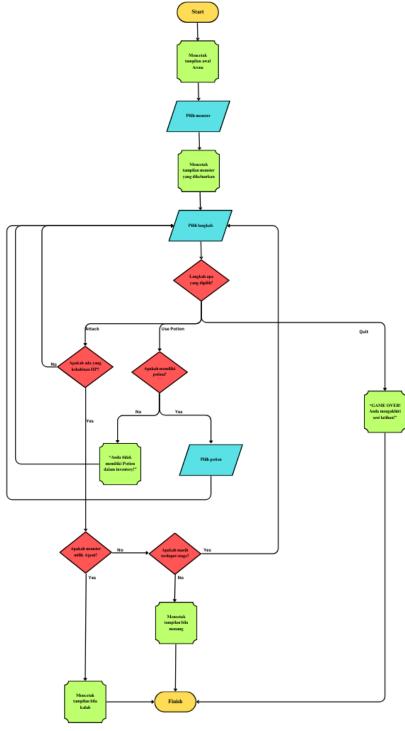


Gambar 7.5 Flowchart Inventory

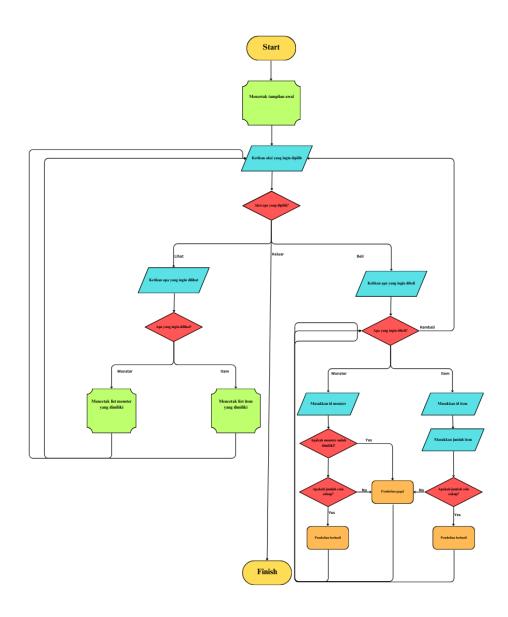
F08 - Battle



F09 - Arena

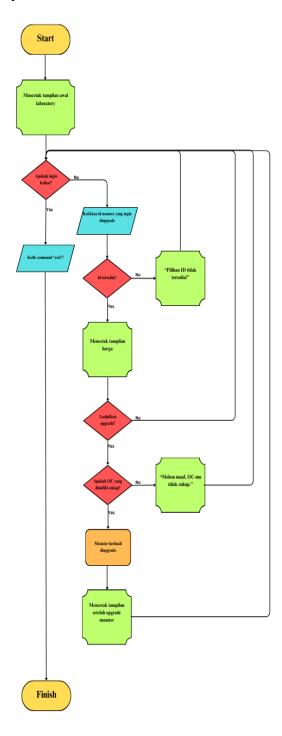


F10 - Shop Currency



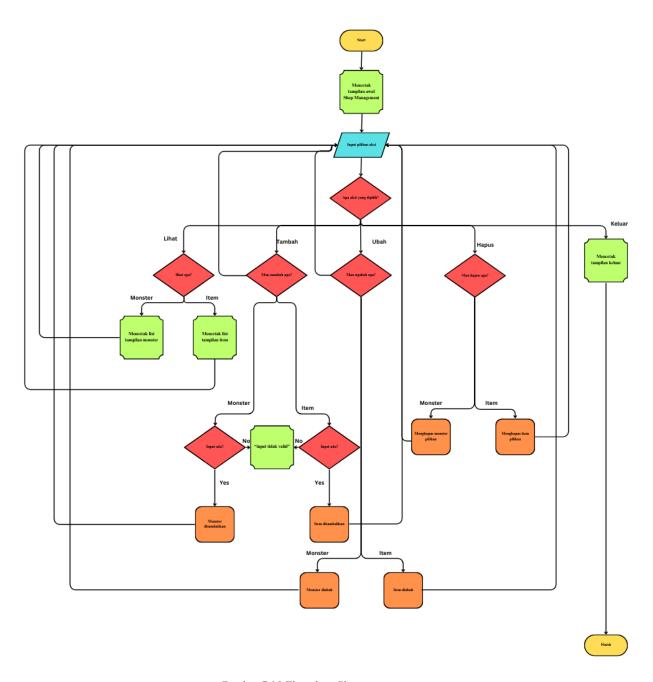
Gambar 7.8 Flowchart Shop dan Currency

F11 - Laboratory



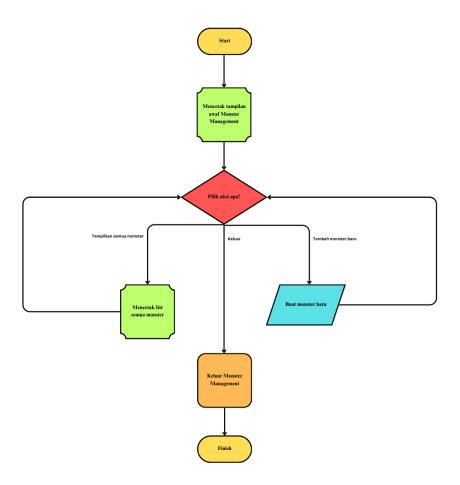
Gambar 7.9 Flowchart Laboratory

F12 - Shop Management



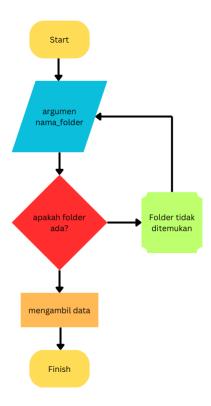
Gambar 7.10 Flowchart Shop management

F13 - Monster Management



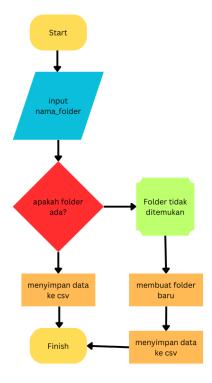
Gambar 7.11 Flowchart Monster Management

F14 - Load



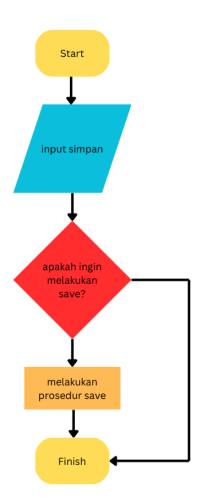
Gambar 7.12 Flowchart Load

F15 - Save



Gambar 7.13 Flowchart Save

F16 - Exit



Spesifikasi Program

```
F00 - Random Number Generator
KAMUS LOKAL
a : int {pengganda}
c : int {peningkatan}
m : int {modulus}
ALGORITMA
import time
    a \leftarrow 1664525
    c \leftarrow 1013904223
    m \leftarrow 2**(32)
    {Fungsi lcg}
         if n = 0 then
              \rightarrow int(time.time()*1000)
         else
               \rightarrow (a*x(n-1) +c)% m
    \rightarrow x(n)
F01 - Register
Prosedur untuk mendaftar akun baru pada game
KAMUS LOKAL
user id : string
username : string
```

```
password
            : string
             : boolean
success
list monster : array of string
monster id : string
ALGORITMA
import src.Constant as c
import os
    if c.data login ≠ ['','','',''] then
        output ('Register gagal!Anda telah login dengan username
        {username}, silahkan lakukan "LOGOUT" sebelum melakukan
        login kembali.')
    else {User belum login, memasukkan data user baru}
        input(user id)
        input(username)
        input(password)
        success ← True
        {Memeriksa kesesuaian karakter dalam username}
        char traversal [username]:
            if not char in
'qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM1234567890
 -' then
                  output ('Username hanya boleh berisi alfabet,
                  angka, underscore, dan strip!')
                  success ← False
                  break
        {Memeriksa ketersediaan username}
        if success then
            row traversal [1..c.data user]:
                if row_1 = username then
                        output('Username {username} sudah
                        terpakai, silahkan gunakan username
                        lain!')
                        success ← False
                        break
        {Menambahkan data user ke dalam file csv, user login}
        if success then
             c.data user = c.data user +
              [[user id, username, password, 'agent', 0]]
             c.data login ← c.data user<sub>(-1)</sub>
```

```
{User memilih monster}
             output('''
             Silahkan pilih salah satu monster sebagai monster
             awalmu.
             1. Pikachow
             2. Bulbu
             3. Zeze
             4. Zuko
             5. Chacha
             ''')
             list monster ←
             ['Pikachow', 'Bulbu', 'Zeze', 'Zuko', 'Chacha']
             while True do
                  monster id ← input('Monster pilihanmu: ')
                  if not monster id in
                   ['1','2','3','4','5']then
                        output('\n\033[31mPilihan nomor tidak
                        tersedia!\033[0m\n')
                   else {Menambahkan data monster user}
                        os.system('cls')
                        c.data monster inventory =
                        c.data monster inventory +
                        [[user id, monster id, '1']]
                        {User berhasil mendaftar}
                        output(f'''
                        Selamat datang Agent {username}. Mari
                        kita mengalahkan Dr. Asep Spakbor dengan
                        {list monster int (monster id) -1 }!
                        111)
                        break
F02 - Login
{Prosedur login untuk mengakses akun user}
KAMUS LOKAL
username : string {username untuk login}
password : string {password}
success : boolean {True jika password benar}
exists : boolean {True jika username ada}
ALGORITMA
```

```
import src.Constant as c
{Memeriksa status login user}
if c.data login ≠ ['','','','']then
     output ('Login gagal!Anda telah login dengan username
      {c.data login[1]}, silahkan lakukan "LOGOUT" sebelum
     melakukan login kembali.')
else {User belum login}
     input(username)
     input (password)
     success ← True
      exists ← False
      {Memeriksa kesesuaian username dan password}
      i traversal [1..(len(c.data user))]:
           if c.data user_{(i)}(1) = username then
                exists ← True
                if c.data user<sub>[i][2]</sub> ≠ password:
                     success ← False
                index user ← i
                break
     if not exists then
           output('Username tidak terdaftar!')
      elif not success then
           output('Password salah')
      else {Username dan password sesuai (valid)}
           {Mengubah data ke-2 di file csv menjadi data user
           yang login}
           c.data login ← c.data_user<sub>index user</sub>
           output('Selamat datang, {role}{username}!Ketik
           command "HELP" untuk menampilkan daftar command yang
           dapat kamu panggil.')
F03 - Logout
{Prosedur logout dari akun user}
KAMUS LOKAL
ALGORITMA
import src.Constant as c
{jika user belum login}
if c.data login = ['','','','']then
```

```
output('Logout gagal!\nAnda belum login, silahkan login
     terlebih dahulu sebelum logout')
{jika user sudah login}
else
     c.data login ← ['','','','']
     output ("Anda telah logout, ketik 'HELP' untuk melihat
     daftar command yang dapat dipanggil." )
F04 - Menu dan Help
{Prosedur menampilkan command yang valid untuk user}
KAMUS LOKAL
ALGORITMA
import src.Constant as c
{jika user belum login}
if c.data login = ['','','',''] then
     output(c.help header)
     output('''
     Kamu belum login sebagai role apapun. Silahkan login
     terlebih dahulu.
     1. login: Masuk ke dalam akun yang sudah terdaftar
     2. register: Membuat akun baru
     3. exit: Keluar
     Footnote:
     Untuk melanjutkan game, silahkan ketik command yang
     terdaftar.
     Jangan lupa untuk memasukkan input yang valid!
      ''')
{jika user login sebagai agent}
elif c.data login<sub>3</sub> = 'agent':
     output(c.help header)
     output(f'''
     Halo Agent {c.data login<sub>1</sub>}. Kamu memanggil command HELP.
     Kamu memilih jalan yang benar, semoga kamu tidak sesat
     kemudian. Berikut adalah hal-hal yang dapat kamu lakukan
     sekarang:
     1. w
                     : Bergerak ke atas
     2. s
                     : Bergerak ke bawah
     3. d
                      : Bergerak ke kanan
     4. a
                     : Bergerak ke kiri
```

```
5. logout
                   : Keluar dari akun yang sedang digunakan
     6. monster
                     : Melihat owca-dex
                    : Melihat inventory
     7. inventory
     8. battle
                    : Melawan monster secara random
     9. arena
                    : Melawan lima monster dengan level 1-5
     10. shop
              : Melihat dan membeli item/monster
     11. laboratory : Menaikkan level monster
     12. jackpot : Main slot
     Footnote:
     Untuk melanjutkan game, silahkan ketik command yang
     terdaftar.
     Jangan lupa untuk memasukkan input yang valid!
     111)
{jika user login sebagai admin}
elif c.data login<sub>3</sub> = 'admin':
     output(c.help header)
     output('''
     Selamat datang, Admin. Berikut adalah hal-hal yang dapat
     kamu lakukan:
     1. logout : Keluar dari akun yang sedang digunakan
     2. shop : Melakukan manajemen pada shop
     3. monster : Melakukan manajemen pada database monster
     Footnote:
     Untuk melanjutkan game, silahkan ketik command yang
     terdaftar.
     Jangan lupa untuk memasukkan input yang valid!
F05 - Monster
{Prosedur menampilkan OWCA DEX yang dimiliki user, fungsi
membaca level monster untuk user, dan prosedur update level
monster untuk user}
KAMUS LOKAL
    index monster : integer
    list monster : array of string
ALGORITMA
```

import src.Constant as c

output(c.owcadex header)

procedure monster

```
output('======= OWCA DEX =======')
      i traversal[1..(len(c.data monster inventory))]
            if (c.data monster inventory[i][0] = c.data login] or
            i = 0 then
                  if i = 0 then
                        index monster \leftarrow 0
                  else
                        index monster ←
                        int(c.data monster inventory[i][])
                  list monster ←
                  [c.data_monster<sub>[index monster][0]</sub>,c.data_monster<sub>[index mon</sub>
                  ster][1], c.data_monster[index monster][2], c.data_monster[i
                  ndex monster][3], c.data_monster[index monster][4], c.data_mon
                  ster_inventory[i][2]]
                  if i \neq 0 and int(list monster<sub>[-1]</sub>)>1 then
                        j traversal [2..5]
                              list monster, ←
                              str(int(int(list monster;) *(int(list m
                              onster<sub>[-1]</sub>)-1)*0.1+int(list monster<sub>i</sub>)))
                 output(f'{list monster<sub>0</sub>:<4}|'+f'{list monster<sub>1</sub>:<1
                 0}|'+f'{list monster<sub>2</sub>:<10}|'+f'{list monster<sub>3</sub>:<10
                  |'+f'| {list monster<sub>4</sub>:<10}|'+f'{list monster<sub>5</sub>:<5}'
function read monster
      row traversal [1..c.data monster inventory]
            if row_0 = id and row_1 = monster id:
                  → row<sub>2</sub>
      → '0'
procedure update monster
      if read monster(id, monster id) ≠ '0'then
            i traversal[1..len(c.data monster inventory))]
                  if c.data monster inventory[i][0] = id and
                  c.data monster inventory [i] = monster id then
                        c.data monster inventory[i][2] ←
                        str(int(c.data monster inventory[i][2]) + chang
                        e lvl)
      else
             c.data monster inventory = c.data monster inventory
             + [[id, monster id, str(change lvl)]]
```

```
F06 - Potion
{Fungsi membaca kuantitas potion/item untuk user dan fungsi
update kuantitas potion/item untuk user}
KAMUS LOKAL
ALGORITMA
import src.Constant as c
function read potion
      row traversal [1..c.data item inventory]
           if row_0 = id and row_1 = type potion then
                \rightarrow row<sub>2</sub>
      → '0'
procedure update potion
      if read potion(id, type potion) ≠ '0' then
           i traversal [1..len(c.data item inventory)]
                if c.data item_inventory[i][0] = id and
                c.data item inventory [i] = type potion then
                      c.data item inventory[i][2] ←
                      str(int(c.data item inventory[i][2])+change_q
                     ty)
                     if read potion(id, type potion) = '0' then
                           c.data item inventory.pop(i)
      else
           c.data item inventory = c.data item inventory +
           [[id, type potion, str(change qty)]]
F07 - Inventory
{Prosedur melihat inventory untuk user}
KAMUS LOKAL
list inventory: array of array of string
monster id : integer
num
               : integer
id
               : string
ALGORITMA
import src. Utility as u
import src.Constant as c
import os
procedure inventory
      output(c.inventory header)
```

```
while True do
     output (f'''
     ======= User ID: {c.data login[0]} ========
     Jumlah O.W.C.A. Coin-mu sekarang adalah
     {c.data login[4]} OC''')
     list inventory ← []
     num \leftarrow 1
           i traversal
           [1..(len(c.data monster inventory))]
           if c.data monster inventory [i][0] = c.data login
           then
                monster id ←
                 int(c.data monster inventory[i][])
                 list inventory = list inventory +
                 [['monster',c.data monster_monster_{id}][1],c.data
                 _monster<sub>[monster id][2]</sub>, c.data_monster<sub>[monster id][3]</sub>,
                 c.data_monster<sub>[monster id][4]</sub>, c.data_monster_inv
                 entory<sub>[i][2]</sub>]]
                 if int(list inventory[-1][5])>1 then
                      j traversal [2..5]
                            list inventory [-1][i] ←
                            str(int(int(list inventory[-1][i]) *
                            (int(list inventory [-1][5]) -1) *0.1+
                            int(list inventory[-1][j])))
                 output(f'{num}. Monster
                                                   (Name:
                 {list inventory[-1][1]}, Lvl:
                 {list inventory<sub>[-1][5]</sub>}, HP:
                 {list inventory[-1][4]})')
                 num = num + 1
     j traversal [1..(len(c.data item inventory))]
           if c.data item inventory [i] [0] = c.data login
           then
                 if c.data item inventory[j][1] ≠ 'monster
                 ball' then
                       output(f'{num}. Potion
                       {c.data item inventory[i][1]}, Qty:
                       {c.data item inventory[i][2]})')
                       list inventory +=
                       [['potion',c.data item inventory[i][1],c
                       .data item_inventory[j][2]]
```

```
else
                      output(f'{num}. Monster Ball (Qty:
                      {c.data item inventory[i][2]})')
                      list inventory += [['monster
                      ball',c.data item inventory[j][2]]]
                 num = num + 1
      {bagian menampilkan detail item atau monster}
     input(id)
     if id = 'exit' then
           os.system('cls')
           break
     elif not u.check int(id) or not int(id) in [1..num]
     then
           os.system('cls')
           output('Pilihan nomor tidak tersedia!')
else
     if list_inventory[int(id)-1][0] = 'monster' then
           os.system('cls')
           output(f'''
           Monster
                   : {list inventory<sub>[int(id)-1][1]</sub>}
           ATK Power : {list inventory[int(id)-1][2]}
           DEF Power : {list inventory[int(id)-1][3]}
                  : {list inventory<sub>[int(id)-1][4]</sub>}
                     : {list inventory[int(id)-1][5]}
           Level
           111)
     elif list inventory [int(id)-1][0] = 'potion' then
           os.system('cls')
           output (f'''
           Potion
                   : {list inventory<sub>[int(id)-1][1]</sub>}
           Type
           Quantity : {list inventory[int(id)-1][2]}
           111)
     elif list_inventory<sub>[int(id)-1][0]</sub> = 'monster ball' then
           os.system('cls')
           output(f'''
           Monster Ball
           Quantity : {list inventory[int(id)-1][2]}
           111)
```

```
F08 - Battle
{Prosedur battle}
KAMUS LOKAL
list_monster
                      : array of string
id musuh
                     : integer
monster id
                      : integer
                      : array of string
stats musuh
                    : array of string
: string
stats agent
lvl agent
hp terakhir
                       : string
attack result : float
strength potion usage : boolean
resilience potion usage : boolean
healing_potion_usage : boolean
                       : boolean
kabur
dmg_multiplier : integer item_exists : boolean
turn
                       : integer
reward
                       : integer
ALGORITMA
import src. Utility as u
import src.Random Number Generator as rnq
import src.Potion as p
import src. Shop Currency as sc
import src.Constant as c
import os
procedure battle
     data monster musuh ← [[c.data monster[i][j] j
     traversal[1..5]] i traversal [1..len(c.data monster))]]
     data monster agent ← [[c.data monster[i][i]] j
     traversal[1..5]] traversal i
     traversal(len(c.data monster))]
     os.system('cls')
     {bagian muncul monster musuh}
     id musuh \leftarrow rng.lcg x(10)%(len(data monster musuh)-1) +1
     lvl musuh \leftarrow rng.lcg x(10)%5 +1
     stats musuh \leftarrow data monster musuh<sub>id musuh</sub>
```

```
if int(lvl musuh)>1 then
      i traversal[2..len(stats musuh)]
           stats musuh; ←
           str(int(int(stats musuh<sub>i</sub>)*(lvl musuh-1)*0.1+int(
           stats musuh;)))
if lvl musuh = 5 then
      output(c.monster level5)
else
     output(c.monster)
output (f'''
RAWRRR, monster {stats musuh<sub>1</sub>} telah muncul
       : {stats musuh<sub>1</sub>}
ATK Power: {stats musuh<sub>2</sub>}
DEF Power : {stats musuh<sub>3</sub>}
           : {stats musuh<sub>4</sub>}
          : {lvl musuh}
Level
''')
{bagian muncul monster user}
output('\n======= MONSTER LIST =======\n')
list monster ← []
num \leftarrow 1
i traversal [1..len(c.data monster inventory)]
     if c.data monster inventory [i][0] = c.data login then
           monster id ← int(c.data monster inventory[i][1])
           output(f'{num}. {data monster agent[monster id][1]}')
           list monster = list monster +
           [[c.data monster inventory[i][2], data monster agen
           t_{[monster id][1]}, data monster agent_{[monster id][2]}, data mon
           ster_agent<sub>[monster id][3]</sub>, data_monster_agent<sub>[monster id][4]</sub>
           num = num + 1
while True:
     input(id agent)
      if not u.check int(id agent) or not int(id agent) in
      [1..num] then
           output('Pilihan nomor tidak tersedia!')
     else
           break
```

```
lvl agent \( \text{list_monster_[int(id agent)-1][0]} \)
stats agent ← list_monster<sub>[int(id_agent)-1]</sub>
max hp \leftarrow stats agent_4
if int(lvl agent)>1 then
      i traversal [2..len(stats agent)]
            stats agent_i \leftarrow
            str(int(int(stats agent<sub>i</sub>)*(int(lvl agent)-1)*0.1
            +int(stats agent<sub>i</sub>)))
if int(lvl agent) = 5 then
      output(c.monster level5)
else
      output(c.monster)
output (f'''
RAWRRR, Agent {c.data login<sub>1</sub>} mengeluarkan monster
{stats agent<sub>1</sub>} !!!
           : {stats musuh<sub>1</sub>}
ATK Power : {stats musuh<sub>2</sub>}
DEF Power : {stats musuh<sub>3</sub>}
ΗP
           : {stats musuh<sub>4</sub>}
Level
         : {lvl musuh}
111)
strength potion usage ← False
resilience potion usage ← False
healing potion usage ← False
kabur ← False
dapat monster ← False
turn ← 1
{bagian setiap turn}
while int(stats agent<sub>4</sub>)>0 and int(stats agent<sub>4</sub>)>0 do
      kabur ← False
      dapat monster ← False
      endturn ← False
      while not endturn do
            output(f'''
            ======= TURN {turn} ({stats agent<sub>1</sub>})
            =========
            1. Attack
```

```
2. Use Potion
3. Use Monster Ball
4. Quit
111)
input(perintah)
if not perintah in ['1','2','3','4'] then
     output('Pilihan nomor tidak tersedia!')
{bagian kabur dari battle}
elif perintah = '4' then
     endturn ← True
     kabur ← True
{bagian monster ball}
elif perintah = '3' then
     item exists \leftarrow False
     monster exists ← False
     j traversal
     [1..len(c.data monster inventory)]
          if c.data monster inventory [i][0] =
          c.data login<sub>0</sub> and
          c.data monster inventory[i][] =
           data_{monster}_{musuh_{[id\ musuh][0]}}then
                monster exists ← True
          j traversal
           [1..len(c.data item inventory)]:
          if c.data item inventory [ ] =
           c.data login, and
          c.data item inventory[j][1] = 'monster
          ball' then
                jumlah item ←
                int(c.data item inventory[i][2])
                item exists ← True
     if monster exists then
          output ('Anda sudah memiliki monster
           {stats musuh<sub>1</sub>} dalam inventory!')
     elif not item exists then
          output ('Anda tidak memiliki Monster
          Ball dalam inventory!')
          else
     else:
           os.system('cls')
```

```
p.update potion(c.data login<sub>0</sub>, 'monster
          ball',-1)
          output ('Swoosshhhhh, Anda
          mengeluarkan Monster Ball !!!')
          number = rng.lcg x(10)%100
          if (lvl musuh = 1 and number < 75) or
           (lvl musuh = 2 and number < 50) or
           (lvl musuh = 3 and number < 25) or
           (lvl musuh = 4 and number < 10) or
           (lvl musuh = 5 and number < 5) then
                os.system('cls')
                c.data monster inventory +=
                [[c.data login<sub>0</sub>,str(id musuh),str
                (lvl musuh)]]
                output('Selamat, Anda berhasil
                mendapatkan monster
                {stats musuh<sub>1</sub>} !!!')
                endturn ← True
                dapat monster ← True
          else
                output ('Yahhh, Anda belum
                berhasil mendapatkan monster
                {stats musuh<sub>1</sub>} !!!')
                output ('Sisa Monster Ball Anda:
                {jumlah item-1}')
{bagian potion}
elif perintah = '2' then
     exists ← False
     j traversal
     [1..len(c.data item inventory)]
          if c.data item inventory[i][0] =
          c.data login<sub>0</sub> and
          c.data item inventory[i][] # 'monster
          ball' then
           exists ← True
```

```
if not exists then
     output('\nAnda tidak memiliki Potion
     dalam inventory!\n')
else
     while True do
          output (f'''
          ====== POTION LIST
           ========
          1. Strength Potion (Qty:
          {p.read potion(c.data login<sub>0</sub>,'str
          ength')}) - Increases ATK Power
          2. Resilience Potion (Qty:
           {p.read potion(c.data login<sub>0</sub>, 'res
          ilience')}) - Increases DEF
          Power
          3. Healing Potion (Qty:
          {p.read potion(c.data login<sub>0</sub>, 'hea
          ling')}) - Restores Health
          4. Cancel
           ''')
          potion ← input('Pilih perintah:
           ')
          if not potion in
           ['1','2','3','4'] then
                output ('Pilihan nomor tidak
                tersedia!')
          elif potion = '1' then
                if
                p.read potion(c.data login<sub>0</sub>,
                'strength') = '0' then
                     output ('Anda tidak
                     memiliki Potion dalam
                     inventory!')
                elif strength potion usage
                = True then
                     output ('Kamu mencoba
                     memberikan ramuan ini
                     kepada {stats agent<sub>1</sub>},
                     namun dia menolaknya
                     seolah-olah dia
                     memahami ramuan
                     tersebut sudah tidak
                     bermanfaat lagi.')
                else
                     os.system('cls')
                     output('Setelah
```

```
meminum ramuan ini,
     aura kekuatan terlihat
     mengelilingi
     {stats agent<sub>1</sub>} dan
     gerakannya menjadi
     lebih cepat dan
     mematikan.')
     stats agent₂←
     str(int(int(stats agen
     t_2) *1.05))
     p.update potion(c.data
     login<sub>0</sub>,'strength',-1)
     strength potion_usage
     ← True
     endturn ← True
     break
elif potion = '2'then
     p.read potion(c.data 1
     ogin<sub>0</sub>, 'resilience') =
     '0' then
           output ('Anda
           tidak memiliki
           Potion dalam
           inventory!')
     elif
     resilience potion usag
     e = True then
           output ('Kamu
           mencoba
           memberikan ramuan
           ini kepada
           {stats agent<sub>1</sub>},
           namun dia
           menolaknya
           seolah-olah dia
           memahami ramuan
           tersebut sudah
           tidak bermanfaat
           lagi.')
     else
           os.system('cls')
           output ('Setelah
           meminum ramuan
           ini, muncul
```

```
sebuah energi
           pelindung di
           sekitar
           {stats agent<sub>1</sub>}
           yang membuatnya
           terlihat semakin
           tangguh dan sulit
           dilukai.')
           stats agent₃ ←
           str(int(int(stats
           agent_{3}) *1.05))
           p.update potion(c
           .data login<sub>0</sub>, 'resi
           lience',-1)
           resilience potion
           usage ← True
           endturn ← True
           break
elif potion = '3' then
p.read potion(c.data login<sub>0</sub>,
'healing') = '0' then
     output('Anda tidak
     memiliki Potion dalam
     inventory!')
elif healing potion usage =
True then
     output ('Kamu mencoba
     memberikan ramuan ini
     kepada {stats agent<sub>1</sub>},
     namun dia menolaknya
     seolah-olah dia
     memahami ramuan
     tersebut sudah tidak
     bermanfaat lagi.')
else
     os.system('cls')
     output('Setelah
     meminum ramuan ini,
     luka-luka yang ada di
     dalam tubuh
     {stats agent<sub>1</sub>} sembuh
     dengan cepat. Dalam
     sekejap, Pikachow
     terlihat kembali prima
```

```
dan siap melanjutkan
                        pertempuran.')
                        stats agent₄ ←
                        str(int(int(stats agen
                        t_4])+int(max hp)*0.25))
                        if int(stats agent<sub>4</sub>]) >
                        int(max hp) then
                              stats agent₄ ←
                              max hp
                        p.update potion(c.data
                         login<sub>0</sub>, 'healing', -1)
                        healing potion usage ←
                        True
                        endturn ← True
                        break
                  elif potion = '4' then
                        break
{bagian attack user}
elif perintah = '1' then
      os.system('cls')
      dmg multiplier \leftarrow rng.lcg x(20)%61+70
     hp terakhir ← stats agent<sub>4</sub>
      attack result ←
      int(stats agent<sub>2</sub>) * ((dmg multiplier) / 10
      0) * ((100-int(stats musuh_3))/100)
      stats agent₄ ←
      str(int(int(stats musuh<sub>4</sub>)-attack resul
      if int(stats agent_4) < 0 then
            stats agent<sub>4</sub> ← '0'
      output (f'''
      SCHWINKKK, {stats agent<sub>1</sub>} menyerang
      {stats agent<sub>1</sub>} !!!
              : {stats agent<sub>1</sub>}
      Name
      ATK Power: {stats agent<sub>2</sub>}
      DEF Power : {stats agent<sub>3</sub>}
                  : {stats agent<sub>4</sub>}
      HP
      Level
                : {lvl musuh}
```

```
Penjelasan: ATT results:
                 {attack result}, HP terakhir ←
                 {hp terakhir}, hasil HP ←
                 {stats musuh<sub>4</sub>}
                 ''')
                 endturn ← True
     if kabur or dapat monster or stats_musuh_4 = '0'
     then
           break
     {bagian attack musuh}
     else
           dmg multiplier \leftarrow rng.lcg x(15)%61+70
           hp terakhir ← stats agent<sub>4</sub>
           attack result ←
           int(stats agent<sub>2</sub>)*((dmg multiplier)/100)*((
           100-int(stats agent_3))/100)
           stats agent₄ ←
           str(int(int(stats agent<sub>4</sub>)-attack result))
           if int(stats agent<sub>4</sub>)<0 then
                 stats agent_4 = '0'
           output (f'''
           ======= TURN {turn} ({stats agent<sub>1</sub>]})
           _____
           SCHWINKKK, {stats agent<sub>1</sub>]} menyerang
           {stats agent<sub>1</sub>} !!!
                  : {stats agent<sub>1</sub>}
           ATK Power: {stats agent<sub>2</sub>}
           DEF Power: {stats agent<sub>3</sub>}
                      : {stats agent<sub>4</sub>}
           ΗP
           Level
                    : {lvl agent}
           Penjelasan: ATT results: {attack result},
           HP terakhir ← {hp terakhir}, hasil HP ←
           {stats agent<sub>4</sub>}
            111)
     turn = turn + 1
{jika kabur dari battle}
```

```
if kabur then
                os.system('cls')
                output('Anda berhasil kabur dari BATTLE!')
           {jika kalah}
          elif stats agent_4 = '0' then
                os.system('cls')
                output ('Yahhh, Anda dikalahkan monster
                {stats musuh<sub>1</sub>}. Jangan menyerah, coba lagi !!!')
           {jika menang}
          elif stats agent<sub>4</sub> = '0' then
                os.system('cls')
                output ('Selamat, Anda berhasil mengalahkan
               monster {stats agent<sub>1</sub>} !!!')
                reward \leftarrow rng.lcg x(10)%26 + 5
                output ('Total OC yang Anda peroleh adalah
                {reward} OC.')
                output('Good luck on your next journey, Agent
                {c.data login<sub>1</sub>}!')
                sc.update oc(reward)
F09 - Arena
{Prosedur arena}
KAMUS LOKAL
                        : array of integer
reward
total hadiah
                        : integer
total win
                        : integer
total_dmg_diberikan : float
list monster
                        : array of string
                        : integer
stage
id musuh
                        : integer
                        : integer
monster id
```

: array of string
: array of string

: string

: string

stats_musuh stats agent

attack_result : float strength_potion_usage : boolean resilience potion usage : boolean

lvl_agent
hp terakhir

```
healing potion usage : boolean
kabur
                           : boolean
dmg multiplier
                         : integer
turn
                          : integer
ALGORITMA
import src. Utility as u
import src.Random Number Generator as rng
import src.Potion as p
import src.Shop Currency as sc
import src.Constant as c
import os
procedure arena
      reward \leftarrow [30,50,100,150,200]
      total hadiah ← 0
      total win \leftarrow 0
      total dmg diberikan \leftarrow 0
      total dmg diterima \leftarrow 0
      {Bagian Setiap Stage}
      stage traversal[1..6]
           data monster musuh ← [[c.data monster[i][i]] j traversal
           [1..5] i traversal[1..len(c.data monster)]]
           data monster agent ← [[c.data monster[i][i]] j traversal
           [1..5] i traversal[1..len(c.data monster)]]
           os.system('cls')
           {bagian muncul monster musuh}
           id musuh \leftarrow rng.lcg x(10)%(len(data monster musuh)-1)
           +1
           stats musuh ← data monster musuh<sub>id musuh</sub>
           if stage>1:
                 i traversal[2..len(stats musuh)]
                      stats musuh; ←
                      str(int(int(stats musuh,)*(stage-1)*0.1+int
                      (stats musuh;)))
           output(f'======= STAGE {stage} ========')
           if stage = 5 then
                 output(c.monster level5)
           else
                 output(c.monster)
```

```
output(f'''
RAWRRR, monster {stats musuh;} telah muncul
       : {stats musuh;}
ATK Power: {stats musuh<sub>2</sub>}
DEF Power : {stats musuh<sub>3</sub>}
          : {stats musuh<sub>4</sub>}
Level
         : {stage}
''')
{bagian muncul monster user}
if stage = 1:
      output('\n====== MONSTER LIST
      =======\n')
      list monster = []
      num \leftarrow 1
      i traversal[1..len(c.data monster inventory)]
            if c.data monster inventory[i][0] =
            c.data login<sub>0</sub> then
                  monster id ←
                  int(c.data monster inventory[i][])
                  output (f'{num}.
                  {data monster agent<sub>[monster id][1]</sub>}')
                  list monster +=
                  [[c.data monster_inventory[i][2], data_mo
                  nster_agent<sub>[monster_id][1]</sub>, data_monster_age
                  nt<sub>[monster id][2]</sub>, data_monster_agent<sub>[monster id]</sub>
                  [3], data_monster_agent[monster_id][4]]
                  num += 1
      while True do
            input(id agent)
            if not u.check int(id agent) or not
            int(id agent) traversal[1..num] then
                  output ('Pilihan nomor tidak
                  tersedia!')
            else
                  break
      lvl agent \leftarrow list_monster<sub>[int(id_agent)-1][0]</sub>
      Stats agent ← list_monster[int(id_agent)-1]
      max_hp \( \text{stats_agent}_4 \)
```

```
else
     id agent traversal [1..len(data monster agent)]
           if data monster_agent[id agent][1] =
           stats agent<sub>1</sub> then
                 stats agent ← data monster agent<sub>id agent</sub>
if int(lvl agent)>1:
     i traversal[2..len(stats agent)]
           stats agent; ←
           str(int(int(stats agent<sub>1</sub>)*(int(lvl agent)-1
           ) *0.1+int(stats agent;)))
if int(lvl agent) = 5 then
     output(c.monster level5)
else
     output(c.monster)
output (f'''
RAWRRR, Agent {c.data login<sub>1</sub>} mengeluarkan monster
{stats agent<sub>1</sub>} !!!
Name
       : {stats agent<sub>1</sub>}
ATK Power: {stats agent<sub>2</sub>}
DEF Power : {stats agent<sub>3</sub>}
         : {stats agent<sub>4</sub>}
        : {lvl agent}
Level
''')
strength potion usage ← False
resilience potion usage ← False
healing potion usage ← False
kabur ← False
turn ← 1
{bagian setiap turn}
while int(stats agent_4)>0 and int(stats musuh_4)>0 do
      endturn ← False
      while not endturn do
           output(f'''
           ======= TURN {turn} ({stats agent<sub>1</sub>})
           =========
           1. Attack
           2. Use Potion
           3. Quit
```

```
''')
perintah ← input('Pilih perintah: ')
if not perintah in ['1','2','3'] then
     output ('Pilihan nomor tidak
     tersedia!')
{bagian kabur dari arena}
elif perintah = '3' then
     endturn ← True
     kabur ← True
{bagian potion}
elif perintah = '2' then
     exists ← False
     i traversal
     [1..len(c.data item inventory)]
     if c.data item inventory[i][0] =
     c.data login<sub>0</sub> then
          exists ← True
     if not exists then
          output ('Anda tidak memiliki
          Potion dalam inventory!')
     else
          while True do
                output (f'''
                ====== POTION LIST
                ========
                1. Strength Potion (Qty:
                {p.read potion(c.data login<sub>0</sub>
                ,'strength')}) - Increases
                ATK Power
                2. Resilience Potion (Qty:
                {p.read potion(c.data login<sub>0</sub>
                ,'resilience')}) -
                Increases DEF Power
                3. Healing Potion (Qty:
                {p.read potion(c.data login<sub>0</sub>
                ,'healing')}) - Restores
                Health
                4. Cancel
                ''')
```

```
potion ← input('Pilih
     perintah: ')
     if not potion in
     ['1','2','3','4'] then
           output('Pilihan nomor
           tidak tersedia!')
           elif potion = '1' then
     if
     p.read potion(c.data login<sub>0</sub>,
     'strength') = '0' then
           output('Anda tidak
          memiliki Potion dalam
           inventory!')
     elif strength potion usage
     = True then
           output ('Kamu mencoba
           memberikan ramuan ini
           kepada {stats agent<sub>1</sub>},
           namun dia menolaknya
           seolah-olah dia
          memahami ramuan
           tersebut sudah tidak
          bermanfaat lagi.')
     else
           os.system('cls')
           output ('Setelah
           meminum ramuan ini,
           aura kekuatan terlihat
          mengelilingi
           {stats agent<sub>1</sub>} dan
           gerakannya menjadi
           lebih cepat dan
          mematikan.')
           stats agent<sub>2</sub>
           str(int(int(stats agen
           t_2) *1.05))
          p.update potion(c.data
           login<sub>0</sub>,'strength',-1)
           strength potion usage
           ← True
           endturn ← True
          break
elif potion = '2' then
```

```
p.read potion(c.data login<sub>0</sub>'
     resilience') = '0' then
           output('Anda tidak
          memiliki Potion dalam
           inventory!')
     elif
     resilience potion usage =
     True then
           output(f'\nKamu
          mencoba memberikan
           ramuan ini kepada
           {stats agent<sub>1</sub>}, namun
           dia menolaknya
           seolah-olah dia
          memahami ramuan
           tersebut sudah tidak
          bermanfaat lagi.\n')
     else
           os.system('cls')
           output(f'\nSetelah
           meminum ramuan ini,
          muncul sebuah energi
           pelindung di sekitar
           {stats agent<sub>1</sub>} yang
           membuatnya terlihat
           semakin tangguh dan
           sulit dilukai.\n')
           stats agent_3 \leftarrow
           str(int(int(stats agen
           t_3) *1.05))
           p.update potion(c.data
           login<sub>0</sub>, 'resilience', -1
           resilience potion usag
           e ← True
           endturn ← True
          break
elif potion = '3' then
     p.read potion(c.data login,
     'healing') = '0' then
           output('Anda tidak
          memiliki Potion dalam
```

```
inventory!')
                      elif healing potion usage =
                      True then
                           output ('Kamu mencoba
                           memberikan ramuan ini
                           kepada {stats agent<sub>1</sub>},
                           namun dia menolaknya
                           seolah-olah dia
                           memahami ramuan
                           tersebut sudah tidak
                           bermanfaat lagi.')
                      else
                           os.system('cls')
                           output(f'\nSetelah
                           meminum ramuan ini,
                           luka-luka yang ada di
                           dalam tubuh
                           {stats agent<sub>1</sub>} sembuh
                           dengan cepat. Dalam
                           sekejap, Pikachow
                           terlihat kembali prima
                           dan siap melanjutkan
                           pertempuran.\n')
                           stats agent[4] ←
                           str(int(int(stats agen
                           t_4) +int (max hp) *0.25))
                           if int(stats agent<sub>4</sub>) >
                           int(max hp) then
                                 stats agent₄ ←
                                 max hp
                           p.update potion(c.data
                            login<sub>0</sub>, 'healing', -1)
                           healing potion usage ←
                           True
                           endturn ← True
                           break
                elif potion = '4' then
                      break
{bagian attack user}
elif perintah = '1' then
     os.system('cls')
     dmg_multiplier \leftarrow rng.lcg_x(20)%61+70
```

```
hp terakhir ← stats musuh<sub>4</sub>
            attack result ←
            int(stats agent<sub>2</sub>) * ((dmg multiplier) /100) * ((
            100-int(stats musuh_3))/100)
            total dmg diberikan +=
            int(stats agent<sub>2</sub>)*((dmg multiplier)/100)*((
            100-int(stats musuh_3))/100)
            stats musuh₄ ←
            str(int(int(stats musuh<sub>4</sub>)-attack result))
            if int(stats musuh<sub>4</sub>)<0 then
                  stats musuh<sub>4</sub> ← '0'
            output (f'''
            SCHWINKKK, {stats agent<sub>1</sub>} menyerang
            \{stats musuh_1\} !!!
            Name
                    : {stats musuh<sub>1</sub>}
            ATK Power: {stats musuh<sub>2</sub>}
            DEF Power : {stats musuh<sub>3</sub>}
                     : {stats musuh<sub>4</sub>}
            HP
            Level : {stage}
            Penjelasan: ATT results: {attack result},
            HP terakhir ← {hp terakhir}, hasil HP ←
            {stats musuh<sub>4</sub>}
            ''')
            endturn ← True
if kabur or stats musuh_4 = '0' then
     break
{bagian attack musuh}
else
      dmg multiplier \leftarrow rng.lcg x(15)%61+70
      hp terakhir ← stats agent<sub>4</sub>
      attack result ←
      int(stats musuh<sub>2</sub>) * ((dmg multiplier)/100) * ((100-i
      nt(stats agent_3))/100)
      total dmg diterima +=
      int(stats musuh<sub>2</sub>) * ((dmg multiplier)/100) * ((100-i
      nt(stats agent_3))/100)
      stats agent_4 \leftarrow
```

```
str(int(int(stats agent<sub>4</sub>)-attack result))
           if int(stats agent_4) < 0 then
                 stats agent₄ ← '0'
           output(f'''
           =======>>> TURN {turn} ({stats agent<sub>1</sub>})
           <<<============
           SCHWINKKK, {stats agent<sub>1</sub>} menyerang
           {stats agent<sub>1</sub>} !!!
                   : {stats agent<sub>1</sub>}
           ATK Power: {stats agent<sub>2</sub>}
           DEF Power: {stats agent<sub>3</sub>}
                     : {stats agent<sub>4</sub>}
           Level : {lvl agent}
           Penjelasan: ATT results: {attack result}, HP
           terakhir ← {hp terakhir}, hasil HP ←
            {stats agent₄}
            ''')
     turn += 1
if kabur:
     break
{jika kalah}
elif stats agent_4 = '0' then
     output ('Yahhh, Anda dikalahkan monster
      {stats musuh<sub>1</sub>}. Jangan menyerah, coba lagi !!!')
     break
{jika menang stage}
elif stats musuh_4 = '0' then
     output ('Selamat, Anda berhasil mengalahkan monster
      {stats musuh<sub>1</sub>} !!!')
     total hadiah += reward<sub>stage-1</sub>
     total win += 1
     output(f'\nSTAGE CLEARED! Anda akan mendapatkan
      {reward<sub>[stage-1]</sub>} OC pada sesi ini!\n')
os.system('cls')
{jika kabur dari arena}
```

```
if kabur then
          output('GAME OVER! Anda mengakhiri sesi latihan!')
     {jika kalah}
     elif total win < 5 then
          output ('GAME OVER! Sesi latihan berakhir pada stage
          {stage}!')
     {jika menang}
     else
          output('\nSelamat, Anda berhasil menyelesaikan
          seluruh stage Arena !!!\n')
     sc.update oc(total hadiah)
     output(f'''
     =======>>> STATS <<<==========
     Total hadiah : {total hadiah} OC
     Jumlah stage : {total win}
     Damage diberikan : {total dmg diberikan}
     Damage diterima : {total dmg diterima}
     ''')
F10 - Shop Currency
{Fungsi update OC untuk user dan prosedur shop}
KAMUS LOKAL
aksi : string
lihat : string
beli : string
id : string
item id : string
monster id : string
jumlah : string
ALGORITMA
import src. Utility as u
import src.Monster as m
import src.Potion as p
import src.Constant as c
import os
procedure update oc
    i traversal[1..len(c.data user)]
```

```
if c.data user[i][0] = c.data login0 then
                                               c.data user_{[i][4]} \leftarrow str(int(c.data user_{[i][4]}) + oc)
                                               break
procedure shop
               {output header}
                output (c.shop header)
                {input aksi user}
                while True do
                                aksi ← input('>>> Pilih aksi (lihat/beli/keluar): ')
                                {bagian lihat}
                                if aksi = 'lihat' then
                                                         input(lihat)
                                                {bagian lihat monster}
                                               if lihat = 'monster' then
                                                               exists ← False
                                                               j traversal [1..len(c.data monster)]
                                                                               if c.data_monster_shop; \( \frac{\capa_1}{\capa_1} \) \( \frac{\ca
                                                                                               exists ← True
                                                                if exists then
                                                                               i traversal[1..len(c.data monster shop)]
                                                                                               if i > 0 then
                                                                                                               id ← c.data monster shop[i][0]
                                                                                               else
                                                                                                               id \leftarrow 0
                                                                                               if id ≠ '' then
                                                                                                                            output(f'{c.data_monster[int(id)][0]
                                                                                                                            :<4} | '
                                                                                                                            +f'{c.data monster[int(id)][1]:<10}|
                                                                                                                            +f'{c.data_monster<sub>[int(id)][2]</sub>:<10}|</pre>
                                                                                                                            +f'{c.data monster[int(id)][3]:<10}|
                                                                                                                            +f'\{c.data monster_{[int(id)][4]}:<10\}
```

```
+f'{c.data monster shop<sub>[i][1]</sub>:<5}|</pre>
                          +f'{c.data monster shop<sub>[i][2]</sub>}')
        else
             output('Tidak ada monster di shop!')
    {bagian lihat item}
    elif lihat = 'item' then
        exists ← False
        j traversal[2..len(c.data item)]:
             if c.data item shop_i \neq ['', '', '', ''] then
                 exists ← True
        if exists then
             i traversal[1..len(c.data item shop)]:
                  if c.data item shop; #
                  ['','','']then
                      output(f'{c.data item shop[i][0]:<5}|'
                          +f'{c.data item shop[;][]:<15}|'
                          +f'{c.data item shop<sub>[i][2]</sub>:<5}|'
                          +f'{c.data item shop[i][3]}')
        else
             output('Tidak ada item di shop!')
    else
        output('Input tidak valid')
{bagian beli}
elif aksi = 'beli' then
    output ('Jumlah O.W.C.A. Coin-mu sekarang
    {c.data login<sub>4</sub>}.')
    while True do
        input(beli)
        if beli = 'kembali' then
             break
         {bagian beli monster}
        elif beli = 'monster' then
             exists ← False
             j traversal[1..len(c.data monster)]:
             if c.data monster shop; \neq ['','',''] then
                      exists ← True
             if exists then
                monster id ← input('Masukkan id monster
                 ')
```

```
if monster id in [f'{j}' j
    traversal[1..len(c.data monster)]] and
    c.data monster shop[int(monster id)] #
    ['','',''] then
             if c.data monster shop[int(monster_id)][1]
             = '0' then
                  output ('Stok monster
                  {c.data monster[int(monster id)][1]}
                  habis! Pembelian dibatalkan.')
             elif
             m.read monster(c.data login, monste
             r id) \neq '0' then
                  output ('Monster
                  {c.data monster[int(monster_id)][1]}
                  sudah ada dalam inventory-mu!
                  Pembelian dibatalkan.')
             elif int(c.data login<sub>4</sub>) <
             int(c.data monster shop[int(monster id)]2)
             then
                  output ('OC-mu tidak cukup.')
             else
                  output(f'Berhasil membeli item
                  {c.data_monster_[int(monster_id)][1]} .
                  Item sudah masuk ke
                  inventory-mu!')
                  update oc(-int(c.data monster s
                  hop<sub>[int(monster id)][2]</sub>))
                  c.data monster_shop[int(monster_id)][1]
                  str(int(c.data monster shop[int(mo
                  nster id)][1])-1)
                  c.data monster inventory +=
                  [[c.data login[0], monster id, '1
                  ' ] ]
                  break
         else
             output ('id monster tidak
             ditemukan')
    else
         output('Tidak ada monster di shop!')
{bagian beli item}
elif beli = 'item' then
    exists ← False
    j traversal[1..len(c.data item)]
        if c.data_item_shop<sub>[j]</sub> \neq ['','','','']
```

```
then
              exists ← True
    if exists then
         input(item id)
         input(jumlah)
         if item id in [f'{j}'j
         traversal[1..len(c.data item shop)] and
         c.data item shop[int(item id)] #
         ['','',''] and u.check int(jumlah)
         then
              if int(c.data item_shop[int(item_id)][2])
              < int(jumlah) or</pre>
              c.data item_shop<sub>[int(item id)][2]</sub> = '0'
              then
                   output(f'\nStok potion
                   {c.data item shop[int(item id)][0]}
                   tidak cukup! Pembelian
                   dibatalkan.\n')
              elif int(c.data login<sub>4</sub>) <
              int(jumlah)*int(c.data item shop[int()]
              item id)][3]) then
                   output(f'\nOC-mu tidak
                   cukup.\n')
              else
                   output(f'Berhasil membeli item:
                   {jumlah}
                   {c.data item shop[int(item id)][1]}.
                   Item sudah masuk ke
                   inventory-mu!')
                   update oc(-int(jumlah)*int(c.da
                   ta item shop[int(item id)][3]))
                   c.data item shop_{[int(item id)][2]} \leftarrow
                   str(int(c.data_item_shop<sub>[int(item_id</sub>
                   )<sub>11[21</sub>) -int(jumlah))
                   p.update potion(c.data login, c.
                   data_item_shop<sub>[int(item id)][1]</sub>, int(ju)
                   mlah))
         else
              output('id item tidak ditemukan')
    else
         output('Tidak ada item di shop!')
else
```

```
output('input tidak valid')
        {keluar}
        elif aksi = 'keluar' then
             os.system('cls')
            output ('Mr. Yanto bilang makasih, belanja lagi ya
            nanti :)')
            break
F11 - Laboratory
{Prosedur laboratory}
import src. Utility as u
import src.Monster as m
import src. Shop Currency as sc
import src.Constant as c
import os
def laboratory() -> None:
KAMUS LOKAL
    list harga : array of integer
    list monster : array of string
    id : string
    lanjutkan : string
ALGORITMA
    list harga \leftarrow [300,500,800,1000]
    {output header}
    output(c.lab header)
    {output monster yang dimiliki user}
    while True do
        output('=======> INVENTORY LIST <========')</pre>
        list monster ← []
        num \leftarrow 1
        i traversal[1..len(c.data monster inventory)]
             if c.data monster inventory[i][0] = c.data_login_0 then
                 index monster ←
                 int(c.data monster inventory[i][])
                 output(f'{num}. {c.data_monster<sub>[index_monster][1]</sub>}
```

```
(Lvl: {c.data monster inventory[i][2]})')
          list monster +=
          [[c.data monster inventory[i][2], c.data_monster[in
          dex monster][1], c.data_monster[index_monster][2], c.data_mon
          ster[index monster][3], c.data_monster[index monster][4]]]
         num += 1
output(f'''
=======>>> UPGRADE PRICE <<<========
Jumlah O.W.C.A. Coin-mu sekarang adalah
{c.data login[4]}
1. Level 1 -> Level 2: 300 OC
2. Level 2 -> Level 3: 500 OC
3. Level 3 -> Level 4: 800 OC
4. Level 4 -> Level 5: 1000 OC''')
{input id monster untuk diupgrade}
input(id)
if id = 'exit' then
     os.system('cls')
    break
{jika pilihan nomor ID tidak tersedia}
elif not u.check int(id) or not int(id)
traversal(1, num) then
     os.system('cls')
     output('Pilihan ID tidak tersedia!')
{jika level monster sudah maksimum}
elif list_monster<sub>[int(id)-1][0]</sub> = '5' then
     os.system('cls')
     output('Maaf, monster yang Anda pilih sudah
    memiliki level maksimum!')
else
     os.system('cls')
     output (f'''
     {\{list\_monster_{[int(id)-1][1]}\}} akan di-upgrade ke level
     {int(list_monster<sub>[int(id)-1][0]</sub>)+1}.
     Harga untuk melakukan upgrade
     {list monster<sub>[int(id)-1][1]</sub>} adalah
     {list harga[int(list\_monster_{[int(id)-1][0])-1]}} OC.
     ''')
     input (lanjutkan)
     while True:
         if lanjutkan = 'y' then
```

```
if int(c.data login<sub>4</sub>) <
                     list harga[int(list monster[int(id)-1][0])-1]
                     then
                          output ('Mohon maaf, OC-mu tidak
                          cukup.')
                          break
                      else
                          i traversal[1..len(c.data monster)]
                          then
                              if c.data monster[i][] =
                              list_monster[int(id)-1][1] then
                                   monster id ← c.data monster[i][0]
                          m.update monster(c.data login, monster i
                          d, 1)
                          sc.update oc(-list harga[int(list monst
                          er_{[int(id)-1][0]})-1])
                          output(f'\nSelamat,
                          {list\_monster_{[int(id)-1][1]}} berhasil
                          di-upgrade ke level
                          {int(list_monster<sub>[int(id)-1][0]</sub>)+1} !\n')
                          break
                 elif lanjutkan = 'n' then
                      os.system('cls')
                      break
                 else
                      output('input tidak valid')
F12 - Shop Management
{Prosedur shop management}
KAMUS LOKAL
aksi : string
lihat : string
tambah : string
ubah : string
hapus : string
id : string
exists : boolean
monster id : string
```

item id : string

```
stok baru : string
harga baru : string
yakin : string
ALGORITMA
import src. Utility as u
import src.Constant as c
import os
procedure shop management
    {output header}
    output(c.shop management header)
    {input aksi user}
    while True:
         input(aksi)
         {bagian lihat}
         if aksi = 'lihat' then
             input(lihat)
              {bagian lihat monster}
              if lihat = 'monster' then
                  exists ← False
                  j traversal[1..len(c.data monster)]:
                       if c.data monster shop; ≠ ['','',''] then
                           exists ← True
                  if exists then
                       i traversal[1..len(c.data monster_shop)]:
                           if i > 0 then
                                id ← c.data monster shop[i][0]
                           else
                                id \leftarrow 0
                           if id \neq '' then
                                  output(f'{c.data monster[int(id)][0]:<</pre>
                                  4}|'+f'{c.data_monster<sub>[int(id)][1]</sub>:<10
                                  }|'+f'{c.data monster[int(id)][2]:<10}</pre>
                                   | '+f'{c.data monster<sub>[int(id)][3]</sub>:<10}|
                                   '+f'{c.data monster[int(id)][4]:<10}|'
                                  +f'{c.data monster shop<sub>[i][1]</sub>:<5}|'</pre>
                                     +f'{c.data monster shop<sub>[i][2]</sub>}')
                  else
                       output('Tidak ada monster di shop!')
```

```
{bagian lihat item}
    elif lihat = 'item' then
        exists ← False
        j traversal[1..len(c.data item)]
             if c.data item shop; \neq ['','','',''] then
                 exists ← True
        if exists then
            i traversal[1..len(c.data item shop)] then
                 if c.data item shop, \neq ['','','','']
                 then
                 output(f'{c.data item shop[i][0]:<5}|'
                 +f'{c.data item shop[i][1]:<15}|'
                 +f'{c.data item shop<sub>[i][2]</sub>:<5}|'
                 +f'{c.data item shop[i][3]}')
       else
            output('Tidak ada item di shop!')
    else
        output('input tidak valid')
{bagian tambah}
elif aksi = 'tambah' then
    while True:
        input(tambah)
        if tambah = 'kembali' then
            break
        {bagian tambah monster}
        elif tambah = 'monster' then
            exists ← False
            j traversal[1..len(c.data monster)]
                 if c.data monster shop; = ['', '', '']
                 then
                     exists ← True
            if exists then
                 i traversal[1..len(c.data monster)]
                       if (i > 0) and not
                       c.data monster[i][0] in
                       c.data monster shop_{[i][0]}) or i = 0
                       then
```

```
output(f'{c.data monster[int(i)]
                [0]:<4}|
                +f'{c.data monster[int(i)][1]:<10
                } | '
                +f'{c.data_monster<sub>[int(i)][2]</sub>:<10
                +f'{c.data_monster<sub>[int(i)][3]</sub>:<10
                } | '
                +f'{c.data monster[int(i)][4]:<10
                }')
    input(monster id)
    input(stok baru)
    input(harga baru)
    if monster id in [f'{j}' j
    traversal[1..len(c.data monster)]] and
    c.data monster shop[int(monster id)] =
    ['','',''] then
          if u.check int(stok baru) and
          u.check int(harga baru) then
                c.data monster_shop[int(monster id)]
                ][0] ← monster id
                c.data monster_shop[int(monster_id)]
                ][2] ← harga baru
                c.data monster_shop[int(monster_id)]
                output(f'{c.data monster[int(mon]
                ster id)][1]} telah berhasil
                ditambahkan ke dalam shop!')
               break
        else
                output('stok dan harga tidak
                valid')
    else
         output ('id monster tidak
         ditemukan')
else
     output ('Semua monster sudah terdapat di
```

```
shop!')
{bagian tambah item}
elif tambah = 'item' then
    exists ← False
    traversal j traversal(1,len(c.data item)):
         if c.data_item_shop; = ['','','','']
         then
             exists ← True
    if exists then
         i traversal[1..len(c.data item)]
               if (i > 0 and not c.data item[i][0]
               in c.data item shop<sub>[i][0]</sub>) or i = 0
               then
                     output(f'{c.data item[int(i)][0]:
                     <4}|
                     +f'{c.data item[int(i)][1]}')
         input(item id)
         input(stok baru)
         input(harga baru)
         if item id in [f'{j}' j
         traversal[1..len(c.data item)]] and
         c.data item shop[int(item_id)] =
         ['','',''] then
               if u.check int(stok baru) and
               u.check int(harga baru) then
                  c.data item_shop_[int(item_id)][0] \leftarrow
                  item id
                  c.data_item_shop_[int(item_id)][2] \leftarrow
                  harga baru
                  c.data item_shop_[int(item_id)][1] \leftarrow
                  stok baru
                  output(f'\n{c.data_item<sub>[int(item id)</sub>
                  | telah berhasil ditambahkan
                  ke dalam shop!\n')
                  break
```

```
else
                          output('stok atau harga tidak
                          valid')
                 else
                      output('id item tidak ditemukan')
             else
                 output ('Semua item sudah terdapat di
                 shop!')
         else
             output('input tidak valid')
{bagian ubah}
elif aksi = 'ubah' then
    while True do
         input(ubah)
        if ubah = 'kembali' then
             break
         {bagian ubah monster}
        elif ubah = 'monster' then
             exists ← False
             j traversal[1..len(c.data monster)]
                  if c.data monster shop; ≠ ['','','']
                  then
                      exists ← True
             if exists then
                  i traversal
                  [1..len(c.data monster shop)]
                      if i > 0 then
                           id ← c.data monster_shop[i][0]
                      else
                          id \leftarrow 0
                      if id ≠ '' then
                             output(f'{c.data_monster[int(id)]
                             1[0]:<4}|
                             +f'{c.data monster[int(id)][1]:<1
                             0}|'
                             +f'{c.data monster[int(id)][2]:<1</pre>
                             0}|'
                             +f'{c.data_monster<sub>[int(id)][3]</sub>:<1</pre>
```

```
0}|
           +f'{c.data monster<sub>[int(id)][4]</sub>:<1</pre>
           0}|'
           +f'{c.data monster shop<sub>[i][1]</sub>:<</pre>
           5}|'
           +f'{c.data monster shop[i][2]}'
input(monster id)
input(stok baru)
input(harga baru)
if monster id in [f'{j}' j
traversal[1..len(c.data monster shop)]]
and c.data monster shop[int(monster id)] #
['','',''] then
      if stok baru = '' and
     u.check int(harga baru) then
           c.data monster_shop[int(monster id)]
           <sub>1[2]</sub> ← harga baru
           output('{c.data_monster[int(mons
           ter_id)][1]} telah berhasil
           diubah dengan harga baru
           {harga baru}!')
           break
     elif u.check int(stok baru) and
     harga baru = '' then
           c.data monster_shop[int(monster_id)]
           output(f'\n{c.data monster[int()]
           monster id)][1]} telah berhasil
           diubah dengan stok baru
           sejumlah {stok baru}')
           break
      elif u.check int(stok baru) and
      u.check int(harga baru) then
        c.data monster_shop[int(monster_id)][2]
```

```
← harga baru
                  c.data monster_shop<sub>[int(monster_id)][1]</sub>
                  ← stok baru
                  output(f'\n{c.data monster<sub>[int(mons</sub>
                  ter id)][1]} telah berhasil diubah
                  dengan stok baru sejumlah
                  {stok baru} dan dengan harga
                  baru {harga baru}!\n')
                  break
             else
                  output ('stok dan harga tidak
                  valid')
         else
             output('id monster tidak
             ditemukan')
    else
         output('Tidak ada monster di shop!')
{bagian ubah item}
elif ubah = 'item' then
    exists ← False
    j traversal[1..len(c.data item)] then
         if c.data item shop; \neq ['','','','']
         then
             exists ← True
    if exists then
         i traversal [1..len(c.data item shop)]:
         if c.data item shop, \neq ['','','','']
         then
               output(f'{c.data item shop[i][0]:<</pre>
               5} | '
               +f'{c.data item shop[i][1]:<15}|'
               +f'{c.data item shop<sub>[i][2]</sub>:<5}|'
               +f'{c.data item shop<sub>[i][3]</sub>}')
         input(item id)
         input(stok baru)
         input(harga baru)
```

```
if item id in [f'{j}' j
         traversal[1..len(c.data item shop)]]
         and c.data item shop[int(item id)] #
         ['','',''] then
               if stok baru = '' and
               u.check int(harga baru) then
                  c.data item shop_{[int(item\_id)][3]} \leftarrow
                  harga baru
                  output(f'\n{c.data item shop[int(
                  item id)][0]} telah berhasil diubah
                  dengan harga baru
                  {harga baru}!\n')
                  break
               elif u.check int(stok baru) and
               harga baru = '' then
                  c.data item shop_{[int(item\_id)][2]} \leftarrow
                  stok baru
                  output ('c.data item shop [int (item id
                  )][0]} telah berhasil diubah
                  dengan stok baru sejumlah
                  {stok baru}')
                  break
               elif u.check int(stok baru) and
               u.check int(harga baru) then
                  c.data item_shop_[int(item_id)][3] \leftarrow
                  harqa baru
                  c.data item\_shop_{[int(item\_id)]2} \leftarrow
                  stok baru
                  output('{c.data item_shop<sub>[int(item]</sub>
                  id) 10} telah berhasil diubah
                  dengan stok baru sejumlah
                  {stok baru} dan dengan harga
                  baru {harga baru}!')
                  break
             else
                     output ('stok dan harga tidak
                     valid')
         else
             output('id item tidak ditemukan')
    else
         output('Tidak ada item di shop!')
else
    output('input tidak valid')
```

```
{bagian hapus}
elif aksi = 'hapus' then
    while True do
         input (hapus)
         if hapus = 'kembali' then
              break
          {bagian hapus monster}
         elif hapus = 'monster' then
              exists ← False
              j traversal[1..len(c.data monster)]:
                    if c.data_monster shop; ≠ ['','','']
                    then
                        exists ← True
              if exists then
                    i traversal
                    [1..len(c.data monster shop)]:
                        if i > 0 then
                             id ← c.data monster_shop[i][0]
                        else
                             id \leftarrow 0
                        if id ≠ '' then
                                output(f'{c.data_monster<sub>[int(id)</sub>
                                ][0]:<4}|
                                +f'{c.data monster[int(id)][1]:<1</pre>
                                0}|'
                                +f'{c.data monster[int(id)][2]:<1</pre>
                                0}|'
                                +f'{c.data monster<sub>[int(id)][3]</sub>:<1</pre>
                                0}|'
                                +f'{c.data monster<sub>[int(id)][4]</sub>:<1</pre>
                                0}|'
                                +f'{c.data monster shop<sub>[i][1]</sub>:<</pre>
                                5}|'
                                +f'{c.data monster shop<sub>[i][2]</sub>}'
```

```
input(monster id)
         if monster id in [f'{j}' j
         traversal[1..len(c.data monster)]] and
         c.data monster shop[int(monster id)] #
         ['','',''] then
             while True do
                 input (yakin)
                 if yakin = 'y' then
                      c.data monster_shop[int(monster_
                     id)]=['','','']
                     output(f'{c.data_monster[int(
                     monster id)][1]} telah berhasil
                     dihapus dari shop')
                     break
                 elif yakin = 'n' then
                     break
                 else
                     output('input tidak valid')
        else
             output ('id monster tidak
             ditemukan')
    else
        output('Tidak ada monster di shop!')
{bagian hapus item}
elif hapus = 'item' then
    exists ← False
    j traversal[1..len(c.data item)]:
         if c.data_item_shop, \neq ['','','']
         then
             exists ← True
    if exists then
        i traversal[1..len(c.data item shop)]:
              if c.data item shop₁ ≠
              ['','','', ''] then
                   output(f'{c.data item shop[i][0]
                   1:<5} | '
                   +f'{c.data item shop<sub>[i][1]</sub>:<15}
```

```
+f'{c.data item shop[i][2]:<5}|
                                     +f'{c.data item shop[i][3]}')
                          input(item id)
                          if item id in [f'{j}' j
                          traversal[2..len(c.data_item)] and
                          c.data item shop[int(item id)] #
                          ['','',''] then
                              while True do
                                 input(yakin)
                                 if yakin = 'y' then
                                       c.data_item_shop<sub>[int(item id)]</sub> 
                                       ['','','','']
                                       output(f'{c.data_item[int(item]
                                       id)][1]} telah berhasil
                                      dihapus dari shop')
                                      break
                                  elif yakin = 'n' then
                                      break
                                  else
                                      output('input tidak valid')
                         else
                              output('id item tidak ditemukan')
                     else
                         output('Tidak ada item di shop!')
                 else
                     output('input tidak valid')
        {keluar}
        elif aksi = 'keluar' then
            os.system('cls')
            output('Dadah Mr. Yanto, sampai jumpa lagi!')
            break
F13 - Monster Management
{Prosedur monster management}
KAMUS LOKAL
aksi : string
```

```
success : boolean
nama : string
attack : string
defense : string
hp : string
tambahkan : string
ALGORITMA
import src. Utility as u
import src.Constant as c
import os
procedure monster management
    {output header}
    output(c.monster header)
    {input aksi user}
    while True do
        aksi ← input('Pilih aksi: ')
        if aksi not in ['1','2','3'] then
            output('Pilihan nomor tidak tersedia!')
        {bagian tampilkan monster}
        elif aksi = '1' then
            i traversal[1..len(c.data monster)]:
                  output(f'{c.data monster[i][0]}' +
                  (5-len(c.data monster[i][0])) *'
                  '+'|'+f'{c.data monster[i][1]}' +
                  (20-len(c.data monster[i][1]))*'
                  '+'|'+f'{c.data monster[i][2]}' +
                  (15-len(c.data monster[i][2]) *'
                  '+'|'+f'{c.data monster[i][3]}' +
                   (15-len(c.data monster[i][3]))*'
                   '+'|'+f'{c.data monster[;][4]}')
        {bagian tambah monster}
        elif aksi = '2' then
            output('Memulai pembuatan monster baru')
            while True do
                 success ← True
                 input (nama)
                  row traversal [c.data monster]
                     if row_1 = nama then
                          output('Nama sudah terdaftar, coba
```

```
lagi!')
            success \leftarrow False
    if success then
        break
while True do
    input(attack)
    if u.check int(attack) then
        break
    else
        output ('Masukkan input bertipe Integer,
        coba lagi!')
while True do
    input(defense)
    if u.check int(defense) and 0<=int(defense)<=50
    then
        break
    else
        output('DEF Power harus bernilai 0-50, coba
        lagi!')
while True do
    input(hp)
    if u.check int(hp) then
        break
    else
        output ('Masukkan input bertipe Integer,
        coba lagi!')
output(f'''
Monster baru berhasil dibuat!
Type : {nama}
ATK Power : {attack}
DEF Power : {defense}
HP : {hp}
''')
while True do
    input(tambahkan)
    if tambahkan = 'y' then
        id ← str(len(c.data monster))
        c.data monster +=
        [[id, nama, attack, defense, hp]]
        c.data monster shop += [['','','']]
```

```
output('\nMonster baru telah
                    ditambahkan!\n')
                    break
                elif tambahkan = 'n' then
                    break
                else
                    output('input tidak valid')
        {keluar}
        elif aksi = '3' then
            os.system('cls')
            break
F14 - Load
{Prosedur load data yang akan dipakai untuk bermain}
KAMUS LOKAL
exists : boolean
folder name : string
argparser : argparse.ArgumentParser
args : argparse.Namespace
ALGORITMA
import src. Utility as u
import src.Constant as c
import argparse
import os
procedure load
    {Membuat Argument Parser}
   argparser ← argparse.ArgumentParser()
   argparser.add argument('nama folder', help←'nama folder
   tempat data program tersimpan')
   args ← argparser.parse args()
   folder name ← args.nama folder
    {cek eksistensi folder}
    exists ← False
    file traversal [..os.listdir('data')]:
        if file = folder name then
            exists ← True
    if exists then
        {mengambil data dari folder}
        c.data user ←
        u.read csv(f'data/{folder name}/user.csv')
```

```
C.data monster ←
        u.read csv(f'data/{folder name}/monster.csv')
        c.data monster inventory ←
        u.read csv(f'data/{folder name}/monster inventory.csv')
        c.data item shop ←
        u.read csv(f'data/{folder name}/item shop.csv')
        c.data item inventory ←
        u.read csv(f'data/{folder name}/item inventory.csv')
        c.data monster shop ←
        u.read csv(f'data/{folder name}/monster shop.csv')
        c.data item ←
        [['id','type'],['1','strength'],['2','resilience'],['3'
        ,'healing'],['4','monster ball']]
        c.data login ← ['','','','','']
        → True
    else
        output(f'Folder "{folder name}" tidak ditemukan.')
        → False
F15 - Save
{Prosedur save}
KAMUS LOKAL
    folder name : string
    folder baru : boolean
ALGORITMA
import src. Utility as u
import src.Constant as c
import os
procedure save
    {input nama folder}
    input(folder name)
    try:
        {membuat folder baru}
        os.makedirs(f'data/{folder name}')
        folder baru ← True
    except:
        folder baru ← False
    {jika membuat folder baru}
    if folder baru then
```

```
output (f'''
         Saving...
         Membuat folder data...
         Membuat folder data/{folder name}...
         Berhasil menyimpan data di folder
         data/{folder name}!''')
    else
         output (f'''
         Saving...
         Berhasil menyimpan data di folder
         data/{folder name}!''')
    {menulis ulang csv}
    u.write csv(f'data/{folder name}/user.csv',c.data user)
    u.write csv(f'data/{folder name}/monster.csv',c.data monste
    r)
    u.write csv(f'data/{folder name}/monster inventory.csv',c.d
    ata monster inventory)
   u.write csv(f'data/{folder name}/item shop.csv',c.data item
   u.write csv(f'data/{folder name}/item inventory.csv',c.data
    item inventory)
   u.write csv(f'data/{folder name}/monster shop.csv',c.data m
    onster shop)
F16 - Exit
{Prosedur exit}
KAMUS LOKAL
simpan : string
ALGORITMA
import src.Save as Save
procedure exit
    {jika belum disave}
    if not saved then
        while True:
            input(simpan)
            if simpan = 'y' then
                Save.save()
```

```
break
            elif simpan = 'n' then
                 break
            else
                 output('input tidak valid')
B04 - Jackpot
{Prosedur bermain jackpot}
KAMUS LOKAL
list item : array of string
list value : array of integer
list monster : array of integer
item1 : integer
item2 : integer
item3 : integer
reward : integer
ALGORITMA
import src. Shop Currency as sc
import src.Random Number Generator as rng
import src.Constant as c
import os
procedure jackpot
    list item ← ['Topi', 'Pedang', 'Koin', 'Potion', 'Monster']
    list value \leftarrow [50, 100, 150, 200, 250]
    {output header}
    output(c.jackpot header)
    {mulai bermain}
    while True do
        input(mulai)
        if mulai = 'n' then
            os.system('cls')
            break
        elif mulai = 'y' then
            {jika OC tidak cukup}
            if int(c.data login_4) < 400 then
                  output ('Maaf, anda tidak memiliki cukup OC
                 untuk bermain JACKPOT.')
            else
                os.system('cls')
```

```
sc.update oc(-400)
item1 \leftarrow rng.lcg x(10)%5
item2 \leftarrow rnq.lcq x(15)%5
item3 \leftarrow rng.lcg x(20)%5
output(f'''\nAnda Mendapatkan: \n
$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$
{list item[item1]:<7}|{list item[item2]:<7}|{li
st item[item3]:<7}
$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
111)
{bagian mendapat jackpot}
if item1 = item2 = item3:
   list monster ← []
   j traversal
   [1..len(c.data monster inventory)] then
         if c.data monster inventory [ ] =
         c.data login<sub>0</sub> then
            list monster = list monster +
            [int(c.data monster inventory[i][]])
   if len(list monster) =
   len(c.data monster)-1 then
       output('\nJACKPOT!!! Selamat, Anda
       mendapatkan 5000 OC\n')
       sc.update oc(5000)
   else
       while True do
            monster id ←
            rng.lcg x(10)%(len(c.data monster)
            -1) +1
            if not monster id in list monster
            then
                 c.data monster inventory +=
                 [[c.data login[0],str(monster
                 id),'1']]
                 output('JACKPOT!!! Selamat,
```

```
Anda mendapatkan monster
                                   {c.data monster<sub>[monster id][1]</sub>}.')
                                   break
                {bagian tidak mendapat jackpot}
                else
                    reward ←
                    list_value<sub>[item1]</sub> + list_value<sub>[item2]</sub> + list_value<sub>[ite</sub>
                    sc.update oc(reward)
                    output('O.W.C.A. Coin sebanyak {reward} OC
                    telah ditambahkan ke akun Anda!')
        else
            output('input tidak valid')
F17 - Main
KAMUS LOKAL
saved : boolean
command : string
danville : array of array of string
run game : bool
ALGORITMA
import src. Save as s
import src.Load as 1d
import src. Exit as e
import src.Shop Management as sm
import src.Monster Management as mm
import src.Laboratory as 1b
import src. Shop Currency as sc
import src.Arena as a
import src.Battle as b
import src.Register as r
import src.Login as ln
import src.Logout as lt
import src.Menu Help as h
import src. Inventory as inv
import src.Monster as m
import src. Jackpot as jc
import src.Constant as c
import os
```

```
['*',' ',' ',' ',' ',' P',' ',' X',' X',' ',' ',' ',' *'],
         ['*',' ','X','X','X',' ',' ','A',' ','X',' ','*'],
         ['*',' ',' ',' ',' ',' ',' ',' ',' X',' ','*'],
         ['*',' ',' ','J',' ',' ','X','X','X','X',' ','*'],
         (x,y) \leftarrow (5,5)
run game ← ld.load()
if run game then
  os.system('cls')
  saved ← True
  output(c.main welcome)
  while True:
     command ← input('>>> ')
     if command = 'help' then
        os.system('cls')
        h.help()
     elif command = 'register' then
        os.system('cls')
        r.register()
        saved ← False
     elif command = 'login' then
        os.system('cls')
        ln.login()
     elif command = 'logout' then
        os.system('cls')
        lt.logout()
     elif command = 'exit' then
        os.system('cls')
        e.exit(saved)
        break
     elif command = 'save' then
        os.system('cls')
        s.save()
        saved ← True
```

```
else
    os.system('cls')
    output('Command tidak ditemukan!')
while c.data login<sub>3</sub> = 'admin':
    input (command)
    if command = 'shop' then
        os.system('cls')
        sm.shop management()
        saved ← False
    elif command = 'monster' then
        os.system('cls')
        mm.monster management()
        saved ← False
    elif command = 'help' then
        os.system('cls')
        h.help()
    elif command = 'login' then
        os.system('cls')
        ln.login()
    elif command = 'logout' then
        os.system('cls')
        lt.logout()
        break
    elif command = 'save' then
        os.system('cls')
        s.save()
        saved ← True
    else
        os.system('cls')
        output('Command tidak ditemukan!')
while c.data login<sub>3</sub> = 'agent':
     output('Agent {c.data login<sub>1</sub>} sedang berada di
     posisi: (\{x-1\}, \{y-1\})')
     peta ← ''
     i traversal[1..12]
           peta += '\n'
           j traversal[1..12]
                 if danville[i][i] = 'X' then
                      if (j \neq 11 \text{ and danville}_{[i][j+1]} \neq 'X')
                      or j = 11 then
                           peta += f''
                      else
                            peta += f''
```

```
elif danville[i][j] = '*' then
                if (j \neq 11 \text{ and danville}_{[i][j+1]} \neq '*')
                or j = 11 then
                       peta += f'{danville<sub>[i][i]</sub>}'
                else
                       peta += f'{danville<sub>[i][j]</sub>}'
           elif danville[i][j+1] = 'X' or
           danville_{[i][j+1]} = '*' then
                peta += f'{danville<sub>[i][i]</sub>}'
           else
           peta += f'{danville<sub>[i][i]</sub>}'
output(peta + '\n')
input(command)
if command = 'w' then
      os.system('cls')
      if danville_{[x-1][y]} = ' ' then
            danville_{[x][y]} \leftarrow ' '
            x -= 1
            danville<sub>[x][v]</sub> ← 'P'
      else
      output('Agent {c.data login<sub>1</sub>} tidak bisa
      pindah karena terdapat Obstacle!')
            if (danville_{[x-1][y]} = 'L') then
                  output('Jika ingin memasuki
                  Laboratory, ketik command
                  "Laboratory"')
            elif (danville_{[x-1][y]} = 'S') then
                  output ('Jika ingin memasuki Shop,
                  ketik command "Shop"')
            elif(danville_{[x-1][y]} = 'A') then
                  output('Jika ingin memasuki Arena,
                  ketik command "Arena", Good
                  Luck!')
            elif(danville_{[x-1][y]} = 'J') then
                  output('Jika ingin bermain
                  JackPot, ketik command "Jackpot"')
elif command = 's' then
      os.system('cls')
      if danville_{[x+1][y]} = ' ' then
      danville<sub>[x][y]</sub> ← ' '
      x += 1
      danville_{[x][y]} \leftarrow 'P'
      else
```

```
output(f'\nAgent {c.data login<sub>1</sub>} tidak
           bisa pindah karena terdapat Obstacle!')
           if (danville_{[x+1][y]} = 'L') then
                 output ('Jika ingin memasuki
                 Laboratory, ketik command
                 "Laboratory"')
           elif (danville_{[x+1][v]} = 'S') then
                 output ('Jika ingin memasuki Shop,
                 ketik command "Shop"')
           elif(danville_{x+1})_{y} = 'A') then
                 output ('Jika ingin memasuki Arena,
                 ketik command "Arena", Good
                 Luck!')
           elif(danville_{[x+1][y]} = 'J') then
                 output('Jika ingin bermain
                 JackPot, ketik command "Jackpot"')
elif command = 'd' then
     os.system('cls')
     if danville_{[x][y+1]} = ' ' then
           danville_{[x][y]} \leftarrow '
           y += 1
           danville_{[x][y]} \leftarrow 'P'
     else
           output('Agent {c.data login<sub>1</sub>} tidak
           bisa pindah karena terdapat Obstacle!')
           if (danville_{(x)}, (v+1)) = 'L') then
                 output ('Jika ingin memasuki
                 Laboratory, ketik command
                 "Laboratory"')
           elif (danville_{[x][y+1]} = 'S') then
                 output ('Jika ingin memasuki Shop,
                 ketik command "Shop"')
           elif(danville_{[x][y+1]} = 'A') then
                 output ('Jika ingin memasuki Arena,
                 ketik command "Arena", Good
                 Luck!')
           elif(danville_{[x][y+1]} = 'J') then
                 output ('Jika ingin bermain
                 JackPot, ketik command "Jackpot"')
elif command = 'a' then
     os.system('cls')
     if danville_{[x][y-1]} = ' ' then
           danville<sub>[x][v]</sub> ← ''
           y = 1
           danville<sub>[x][v]</sub> ← 'P'
     else
```

```
output('Agent {c.data login<sub>1</sub>} tidak
           bisa pindah karena terdapat Obstacle!')
           if (danville_{[x][y-1]} = 'L') then
                output ('Jika ingin memasuki
                Laboratory, ketik command
                 "Laboratory"')
           elif (danville_{[x][y-1]} = 'S') then
                output ('Jika ingin memasuki Shop,
                ketik command "Shop"')
           elif (danville_{[x],[v-1]} = 'A') then
                output ('Jika ingin memasuki Arena,
                ketik command "Arena", Good
                Luck!')
           elif(danville_{[x][y-1]} = 'J') then
                output ('Jika ingin bermain
                JackPot, ketik command "Jackpot"')
elif command = 'help' then
     os.system('cls')
     h.help()
elif command = 'login' then
     os.system('cls')
     ln.login()
elif command = 'logout' then
     os.system('cls')
     lt.logout()
break
elif command = 'save' then
     os.system('cls')
     s.save()
     saved ← True
elif command = 'inventory' then
     os.system('cls')
     inv.inventory()
elif command = 'battle' then
     os.system('cls')
     if 'X' in
     [danville_{[x+1][y]}, danville_{[x-1][y]}, danville_{[x][y+1]}, d
     anville [x][v-1] then
           b.battle()
           saved ← False
     else
           output(f'\nAgent {c.data login<sub>1</sub>} tidak
           berada di area battle!\n')
elif command = 'arena' then
```

```
os.system('cls')
     if 'A' in
      [danville_{[x+1][y]}, danville_{[x-1][y]}, danville_{[x][y+1]}, d
      anville[x][y-1] then
           a.arena()
           saved ← False
     else
           output(f'\nAgent {c.data login<sub>1</sub>} tidak
           berada di area arena!\n')
elif command = 'shop' then
     os.system('cls')
     if 'S' in
      [danville_{[x+1][y]}, danville_{[x-1][y]}, danville_{[x][y+1]}, d
     anville[x][v-1] then
           sc.shop()
           saved ← False
     else
           output(f'\nAgent {c.data login<sub>1</sub>} tidak
           berada di area shop!\n')
elif command = 'laboratory' then
     os.system('cls')
     if 'L' in
      [danville_{[x+1][y]}, danville_{[x-1][y]}, danville_{[x][y+1]}, d
     anville_{[x][y-1]}] then
           lb.laboratory()
           saved ← False
     else
           output(f'\nAgent {c.data login<sub>1</sub>} tidak
           berada di area laboratory!\n')
elif command = 'jackpot' then
     os.system('cls')
     if 'J' in
      [danville_{[x+1][y]}, danville_{[x-1][y]}, danville_{[x][y+1]}, d
     anville[x][y-1] then
           jc.jackpot()
           saved ← False
     else
           output('Agent {c.data login<sub>1</sub>} tidak
           berada di area jackpot!\n')
elif command = 'monster' then
     os.system('cls')
```

```
m.monster()
               else
                    os.system('cls')
                    output('Command tidak ditemukan!')
F18 - Utility
{Fungsi menulis ulang csv, fungsi cek string sebagai int
non-negatifungsi membaca file csv sebagai matriks}
KAMUS LOKAL
array: array of string
limit : int
data : array of array of string
ALGORITMA
function split str
    array ← []
    limit ← 0
    i traversal[1..(len(string))]
        if string; = delimiter then
            array += [string<sub>limit:i</sub>]
            limit ← i+1
    array += [string[]imit:]
    → array
function read csv
    data ← []
    with open(file name, 'r') as file:
        line traversal [1..file]
            row = split str(line[:-1],';')
            data += [row]
    → data
procedure write csv
    with open(file name, 'w', newline='') as file:
        row traversal [1..data]
            file.write(';'.join(map(str, row)) + '\n')
function check int
    if len(string) = 0 then
        → False
    elif string<sub>0</sub> not in '123456789' and len(string)>1 then
        → False
    char traversal [1..string]
        if char not in '1234567890' then
            → False
```

→ True

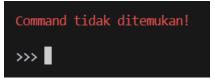
Tabel 4 Notasi Algoritma

Hasil Pengujian Program

Main



Gambar 9.1 Welcome Page



Gambar 9.2 Command Not Found

F01 - Register

```
Masukan username:
Username tidak boleh kosong!
Masukan username: allen
Masukan password:
Password tidak boleh kosong!
Masukan username: halo

Silahkan pilih salah satu monster sebagai monster awalmu.

1. Pikachow
2. Bulbu
3. Zeze
4. Zuko
5. Chacha

Monster pilihanmu:

Pilihan nomor tidak tersedia!

Monster pilihanmu:
```

Gambar 9.3 Register

F02 - Login Username: akhakhakh Password: jddjdj >>> Gambar 9.4 Login - Username tidak terdaftar Username: jatinangor Password: surabaya >>> Gambar 9.5 Login - Password salah Username: Mr_Monogram Password: monogrammer77 Selamat datang, admin Mr_Monogram! Ketik command "HELP" untuk menampilkan daftar command yang dapat kamu panggil. >>> Gambar 9.6 Login - Admin F03 - Logout Logout gagal! Anda belum login, silahkan login terlebih dahulu sebelum logout Gambar 9.7 Logout - Belum Login Anda telah logout, ketik 'HELP' untuk melihat daftar command yang dapat dipanggil. >>> ■ Gambar 9.8 Logout - Berhasil F04 - Menu Help

Gambar 9.9 Menu Help - Sebelum Login

```
Selamat datang, Admin. Berikut adalah hal-hal yang dapat kamu lakukan:

1. logout : Keluar dari akun yang sedang digunakan
2. shop : Melakukan manajemen pada shop
3. monster : Melakukan manajemen pada database monster

Footnote:
Untuk melanjutkan game, silahkan ketik command yang terdaftar.
Jangan lupa untuk memasukkan input yang valid!

>>>> ■
```

Gambar 9.10 Menu Help - Admin

```
Hello Agent jatinanger. Kamu memanggil command HELP. Kamu memilih jalan yang benar, semoga kamu tidak sesat kemudian. Berikut adalah hal-hal yang dapat kamu laku kan sekarang:

1. w : Rengerak ke atas
2. s : Bengerak ke basah
3. d : Bengerak ke basah
4. a : Bengerak ke isin
5. logout : Koluar dari akun yang sedang digunakan
6. moster : Malibati keca dax
6. moster : Milibati inventory
8. battle : Milasan moster secara random
9. stopp : Milibati inventory
10. stopp : Milasan iina moster dengan level 1-5
10. stopp : Milasan iina moster dengan level 1-5
11. laboratory : Menaikkan level moster
12. jackpot : Halin slot
Footrote:
Footr
```

Gambar 9.11 Menu Help - Setelah Login

F05 - Monster

```
Agent jatinangor sedang berada di posisi: (4,4)

/ \ / )(\ / _ ) / _ \ _ ( \( ( _ )( \/ ) \)
( 0 )\ /\ /( ( _ / \( ( _ ))) D ( ) _ ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ ))) D ( ) _ ) ) ) ( \( ( _ / \( ( _ )) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ )) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ )) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ )) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ( \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ) ( \( ( _ / \( ( _ / \( ( _ ) D ( ) D ( ) D ( ) ) ) ) ) ( \( ( _ / \( ( _ / \( ( _ / \( ( _ / \( ( _ / \( ( _ / \( ( _ / \( ( _ / \( ( _ / \( ( _ / \(
```

Gambar 9.12 Monster - OWCA-DEX

F06 - Potion

Potion Type : strength Quantity : 19

Gambar 9.13 Potion - Detail Potion

Gambar 9.13.1 Potion - Detail Potion dalam Pertarungan

F07 - Inventory

Gambar 9.14 Inventory

```
Monster
Name : Pikachow
ATK Power : 137
DEF Power : 11
HP : 660
Level : 2

-----------
Jumlah O.W.C.A. Coin-mu sekarang adalah 9216 OC
1. Monster (Name: Pikachow, Lvl: 2, HP: 660)
2. Monster (Name: Bulbu, Lvl: 1, HP: 1200)
3. Monster (Name: Zuko, Lvl: 1, HP: 800)
4. Monster (Name: Thanos, Lvl: 5, HP: 2800)
5. Potion (Type: strength, Qty: 19)
6. Potion (Type: resilience, Qty: 3)
7. Potion (Type: healing, Qty: 2)
8. Monster Ball (Qty: 2)

Ketikkan nomor ID untuk menampilkan detail item
Jika ingin keluar dari Inventory, ketik command "exit".
```

Gambar 9.15 Inventory - Detail Inventory

F08 - Battle

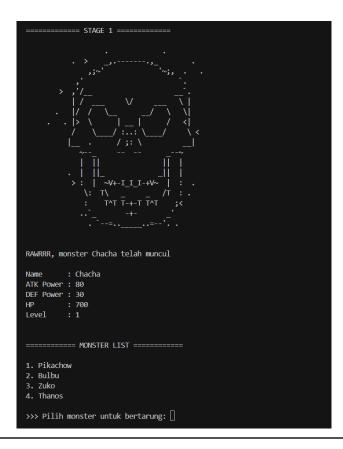
Gambar 9.16 Battle



Gambar 9.17 Battle - Turn

Gambar 9.18 Battle - Results

F09 - Arena



Gambar 9.19 Arena



Gambar 9.20 Arena - Turn

Gambar 9.21 Arena - Potion

Gambar 9.22 Arena - Results - Potion

```
SCHWINKKK, Bulbu menyerang Chacha !!!
       : Chacha
ATK Power: 80
DEF Power : 30
Level
        : 1
Penjelasan: ATT results: 46.9559999999999, HP terakhir = 700, hasil HP = 653
  ----->>> TURN 2 (Chacha) <<<-----
SCHWINKKK, Chacha menyerang Bulbu !!!
        : Bulbu
ATK Power : 52
DEF Power : 50
     : 1113
Level
Penjelasan: ATT results: 38.0, HP terakhir = 1151, hasil HP = 1113
        ==== TURN 3 (Bulbu) ========
1. Attack
2. Use Potion
3. Quit
Pilih perintah:
```

Gambar 9.23 Arena - Results - Attack



Gambar 9.24 Arena - Quit

F10 - Shop Currency

```
.-') ('-... __(-.. (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 ) (00 )
```

Gambar 9.25 Shop - Welcome

Gambar 9.26 Shop - Lihat

```
>>> Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A. Coin-mu sekarang 9216.
>>> Mau beli apa? (monster/item/kembali): monster
Masukkan id monster: 4
Monster Zuko sudah ada dalam inventory-mu! Pembelian dibatalkan.
>>> Mau beli apa? (monster/item/kembali):
```

Gambar 9.27 Shop - Beli Gagal

```
>>> Mau beli apa? (monster/item/kembali): item
>>> Masukkan id item: 4
>>> Masukkan jumlah: 1
Berhasil membeli item: 1 monster ball. Item sudah masuk ke inventory-mu!
>>> Mau beli apa? (monster/item/kembali):
```

Gambar 9.28 Shop - Beli Berhasil

Gambar 9.29 Shop - Keluar

F11 - Laboratory



Gambar 9.30 Lab - Welcome

Gambar 9.31 Lab - Upgrade

Gambar 9.32 Lab - Invalid Input

F12 - Shop Management



Gambar 9.33 Shop Management - Welcome

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): lihat
>>> Mau lihat apa? (monster/item): monster
     |type
|Pikachow
|Bulbu
                   |atk_power |def_power |hp
                                                              |stock|price
                                 10
                                                600
                                                              10
                                                                     |500
                                 50
                    50
                                               1200
                                                                     700
     Zeze
                    300
                                 10
                                                100
                                                                     400
     Zuko
                    100
                                                800
                                                                     550
     Chacha
                                                             |6
|17
                    80
                                 130
                                                700
                                                                     1600
     Godzilla
                                 40
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ihat
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): item
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): lihat
>>> Mau lihat apa? (monster/item): item
      |type
|strength
                           |stock|price
      resilience
                                   30
      |healing
|monster ball
                                   20
                                  500
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar):
```

Gambar 9.34 Shop Management - Lihat

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): tambah
>>> Mau nambah apa? (monster/item/kembali): item
Semua item sudah terdapat di shop!
>>> Mau nambah apa? (monster/item/kembali): monster
id |type
               |atk_power |def_power |hp
   Thanos
                         50
               999
                                    2000
>>> Masukkan id monster: 8
id monster tidak ditemukan
>>> Mau nambah apa? (monster/item/kembali): 7
>>> Mau nambah apa? (monster/item/kembali): monster
               |atk_power |def_power |hp
id |type
                                    2000
    Thanos
>>> Masukkan id monster: 7
>>> Masukkan stok awal: 8
>>> Masukkan harga: 9000
Thanos telah berhasil ditambahkan ke dalam shop!
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar):
```

Gambar 9.35 Shop Management - Tambah

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah
>>> Mau ngubah apa? (monster/item/kembali): item
id
     |type
                     |stock|price
     strength
                     6
                           50
     |resilience
                           130
     |healing
                     8
                           20
     |monster ball
4
                           500
>>> Masukkan id item: 3
>>> Masukkan stok baru: 2
>>> Masukkan harga baru: 90
3 telah berhasil diubah dengan stok baru sejumlah 2 dan dengan harga baru 90!
```

Gambar 9.36 Shop Management - Ubah

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus
>>> Mau hapus apa? (monster/item/kembali): monster
               |atk_power |def_power |hp
                                                |stock|price
id
   |type
    Pikachow
               1125
                          110
                                     1600
                                                10
                                                      1500
    Bulbu
                                                      700
               150
                          150
                                     1200
                                                2
    Zeze
               300
                          10
                                     100
                                                6
                                                      400
    Zuko
               100
                          25
                                     800
                                                19
                                                      550
    Chacha
               80
                          30
                                     700
                                                6
                                                      600
    |Godzilla
               200
                                                17
                                                      1000
                          40
                                     1000
               999
    Thanos
                          |50
                                     2000
                                                      19000
                                                8
>>> Masukkan id monster: 7
>>> Apakah anda yakin ingin menghapus Thanos dari shop (y/n)? y
Thanos telah berhasil dihapus dari shop
>>> Mau hapus apa? (monster/item/kembali):
```

Gambar 9.37 Shop Management - Hapus

```
Dadah Mr. Yanto, sampai jumpa lagi!
```

Gambar 9.38 Shop Management - Keluar

F13 - Monster Management



Gambar 9.39 Monster Management - Welcome

```
SELAMAT DATANG DI DATABASE PARA MONSTER !!!
1. Tampilkan semua Monster
2. Tambah Monster baru
3. Keluar
Pilih aksi: 1
                         |atk_power
                                         |def_power
     Pikachow
                                         10
                                                         600
     Bulbu
                                                         1200
                                         50
                                         10
     Zeze
                          300
                                                         100
     Zuko
                          100
                                         25
                                                         800
     Chacha
                         80
                                         30
                                                         700
     |Godzilla
                                         40
                          200
                                                         11000
     |Thanos
                         999
                                         50
                                                         2000
Pilih aksi:
```

Gambar 9.40 Monster Management - Tampil

```
Pilih aksi: 2

Memulai pembuatan monster baru

Masukkan Type/Nama : gail
Masukkan ATK Power : 500
Masukkan DEF Power (0-50) : 60
DEF Power harus bernilai 0-50, coba lagi!
Masukkan DEF Power (0-50) : 14
Masukkan HP : 200

Monster baru berhasil dibuat!
Type : gail
ATK Power : 500
DEF Power : 14
HP : 200

Tambahkan Monster ke database (y/n) : y

Monster baru telah ditambahkan!
```

Gambar 9.41 Monster Management - Tambah

F14 - Load

```
PS C:\Users\LENOVO\OneDrive - Institut Teknologi Bandung\Semester 2\IF1210\Tubes\if1210-2024-tubes-k07-c> python main.py bla Folder "bla" tidak ditemukan.

PS C:\Users\LENOVO\OneDrive - Institut Teknologi Bandung\Semester 2\IF1210\Tubes\if1210-2024-tubes-k07-c>
```

Gambar 9.42 Load - Error

F15 - Save

```
Apakah Anda mau melakukan penyimpanan file yang sudah diubah? (y/n) y
Masukkan nama folder: save1
Saving...
Berhasil menyimpan data di folder data/save1!
PS C:\Users\LENOVO\OneOrive - Institut Teknologi Bandung\Semester 2\IF1210\Tubes\if1210-2024-tubes-k07-c> ■
```

Gambar 9.43 Save - Existing File Apakah Anda mau melakukan penyimpanan file yang sudah diubah? (y/n) y Masukkan nama folder: halo Saving... Membuat folder data... Membuat folder data/halo... Berhasil menyimpan data di folder data/halo! PS C:\Users\LENOVO\OneDrive - Institut Teknologi Bandung\Semester 2\IF1210\Tubes\if1210-2024-tubes-k07-c> Gambar 9.44 Save - File Baru F16 - Exit PS C:\Users\LENOVO\OneDrive - Institut Teknologi Bandung\Semester 2\IF1210\Tubes\if1210-2024-tubes-k07-c> Gambar 9.45 Exit Apakah Anda mau melakukan penyimpanan file yang sudah diubah? (y/n) y Masukkan nama folder: save1 Saving... Berhasil menyimpan data di folder data/save1! PS C:\Users\LENOVO\OneDrive - Institut Teknologi Bandung\Semester 2\IF1210\Tubes\if1210-2024-tubes-k07-c> Gambar 9.46 Exit - Save B03 - Monster Ball Swoosshhhhh, Anda mengeluarkan Monster Ball !!! Yahhh, Anda belum berhasil mendapatkan monster Chacha !!! Sisa Monster Ball Anda: 2 ======== TURN 1 (Thanos) ========= 1. Attack 2. Use Potion 3. Use Monster Ball 4. Quit

Gambar 9.47 Monster Ball - Gagal

Pilih perintah:



Gambar 9.48 Monster Ball - Berhasil

B04 - Jackpot



Gambar 9.49 Jackpot - Welcome

Gambar 9.50 Jackpot - Result

B05 - Peta Kota Danville

```
Selamat datang Agent allen. Mari kita mengalahkan Dr. Asep Spakbor dengan Pikachow!
Agent allen sedang berada di posisi: (4,4)
>>>
               Gambar 9.51 Peta Kota Danville - Titik Awal
    Agent jafar tidak bisa pindah karena terdapat Obstacle!
Jika ingin memasuki Laboratory, ketik command "Laboratory"
    Agent jafar sedang berada di posisi: (5,8)
                Gambar 9.52 Peta Kota Danville - Obstacle
```

Tabel 5 Hasil Pengujian Program

Lampiran

- 1. Link Lampiran
 - GitHub Repository: <u>link</u>
- 2. Referensi
 - Laporan Asisten: <u>link</u>
 - Github Repository Asisten: <u>link</u>

Form MoM Asistensi Tugas Besar IF1210/Dasar Pemrograman

Sem. 2 2023/2024

Nomor Asistensi

No. Kelompok/Kelas Tanggal asistensi

C/07 3 Mei 2023

Anggota kelompok

	NIM / Nama (Hanya yang Hadir)
1	19623227/Jessica Allen
2	16523217/Faiz Arkan Waskitazaman
3	19623217/Nawaf Amjad Rizqi Aldaha Ismail
4	16523137/Muhammad Jafar Fadli
5	16523007/Mikael Adrian Widagdo
6	
	NIM / Nama

Asisten pembimbing

13521135/Nicholas Liem

Catatan Asistensi:

Rangkuman Diskusi

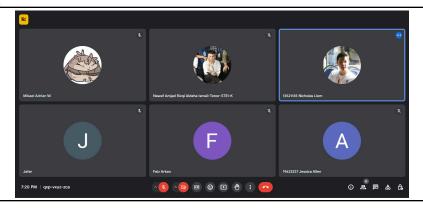
Display: dibebaskan, tidak harus seragam dengan contoh.

Bonus: diusahakan mengerjakan bonus 1 dan 2. Typing python yang dimaksud adalah deklarasi tipe variabel dan fungsi yang digunakan, serta penamaan yang baik dan benar.

Shop Management: tidak perlu membuat database sendiri untuk item (potion dan monster ball)

Tindak Lanjut

Dokumentasi



Form MoM Asistensi Tugas Besar IF1210/Dasar Pemrograman Sem. 2 2023/2024

Nomor Asistensi : 2
No. Kelompok/Kelas : C/07

Tanggal asistensi : 10 Mei 2024

Anggota kelompok

	NIM / Nama (Hanya yang Hadir)
1	19623227/Jessica Allen
2	16523217/Faiz Arkan Waskitazaman
3	19623217/Nawaf Amjad Rizqi Aldaha Ismail
4	16523137/Muhammad Jafar Fadli
5	16523007/Mikael Adrian Widagdo
6	
	NIM / Nama
	40-0440-070 1 2 7 0

Asisten pembimbing

13521135/Nicholas Liem

Catatan Asistensi:

Rangkuman Diskusi

Jangan lupa tiap file tulis header, penulisnya siapa

Ascii dan constant seperti (data[1]) bisa di define "1" nya itu apa, bisa declare di satu tempat, biar ga panjang-panjang di code, mungkin namanya bisa constant biar bisa dipanggil aja, jangan lupa kasi komen deskripsi pada code yang kurang jelas.

utility "U" nya huruf kapital biar konsisten

Pembagian referensi laporan dan repo:

Laporan:

https://drive.google.com/file/d/1mNmClnyCJnVoyuP-gg2Z6s1FdLTxysge/view?usp=sharing

Reno:

https://github.com/NicholasLiem/IF1210 TugasBesar BNMO-Toko-Game

Tindak Lanjut

- declare variable
- memberikan command pada src code
- membuat header setiap file
- manfaatkan bonus 1

Dokumentasi

