

**Laporan Tugas Besar**  
**IF1201-06 Dasar Pemrograman**  
**Program Fungsional Manajerial Candi**



**Kelompok 10 :**

**(16522024) Bagaswara**

**(19622024) Serenada Cinta Sunindyo**

**(19622150) Dedy Hofmanindo Saragih**

**(19622078) Raden Francisco Trianto Bratadiningrat**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**

**INSTITUT TEKNOLOGI BANDUNG**

**2023**

## Halaman Pernyataan

*"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 2022/2023."*

- Bagaswara (16522024)
- Serenada Cinta Sunindyo (19622024)
- Dedy Hofmanindo Saragih (19622150)
- Raden Francisco Trianto Bratadiningrat (19622078)

# Daftar Isi

Halaman Pernyataan.....	2
Daftar Isi.....	3
Daftar Tabel.....	4
Daftar Gambar.....	5
Deskripsi Persoalan.....	6
Daftar Pembagian Kerja Kelompok.....	8
Checklist Hasil Rancangan.....	10
Desain Command.....	12
Desain Kamus Data.....	25
Desain Dekomposisi Algoritmik dan Fungsional Program.....	35
Spesifikasi Program dan Fungsi dalam Notasi Algoritmik.....	47
Hasil Pengujian Program.....	84
Lampiran Form Asistensi.....	89

## Daftar Tabel

Tabel 2.1- Tabel Daftar Pembagian Kerja Kelompok.....	8
Tabel 2.2- Tabel Checklist Hasil Rancangan.....	10

## Daftar Gambar

Gambar 3.1. - Program Login.....	2
Gambar 3.2. - Program Logout.....	3
Gambar 3.3. - Program Summon Jin.....	4
Gambar 3.4. - Program Hilangkan Jin.....	5
Gambar 3.5. - Program Ubah Tipe Jin .....	2
Gambar 3.6. - Program Jin Pembangun.....	3
Gambar 3.7. - Program Jin Pengumpul.....	4
Gambar 3.8. - Program Batch Kumpul/Bangun.....	5
Gambar 3.9. - Program Ambil Laporan Jin.....	2
Gambar 3.10. - Program Ambil Laporan Candi.....	3
Gambar 3.11 - Program Hancurkan Candi.....	4
Gambar 3.12. - Program Ayam Berkokok.....	5
Gambar 3.13. - Program Load.....	2
Gambar 3.14. - Program Save.....	3
Gambar 3.15. - Program Help.....	4
Gambar 3.16. - Program Exit.....	5
Gambar 3.17. - Asistensi Pertama.....	4
Gambar 3.18. - Asistensi Kedua.....	5

# Deskripsi Persoalan

Dalam Tugas Besar IF1210 tahun akademik 2022/2023, kita diminta untuk mengembangkan sebuah program yang menyerupai gim yang didasarkan pada cerita rakyat "Roro Jonggrang dan Bandung Bondowoso". Cerita rakyat ini menceritakan tentang kisah cinta antara Bandung Bondowoso dan Roro Jonggrang. Konflik dalam cerita ini adalah lamaran Bandung Bondowoso yang selalu ditolak oleh Roro Jonggrang. Pada akhirnya, Roro Jonggrang tidak tahan lagi melihat usaha Bandung Bondowoso yang berusaha merayu hatinya, sehingga Roro Jonggrang memberikan tantangan kepada Bandung Bondowoso yaitu agar Bandung Bondowoso membangun 100 candi dalam waktu semalam sebagai syarat agar Bandung Bondowoso dapat diterima sebagai pasangan Roro Jonggrang. Tanpa berpikir panjang, Bandung Bondowoso yang sangat cinta kepada Roro Jonggrang langsung menerima tantangan yang diberikan oleh Roro Jonggrang tersebut dan meminta bantuan pada jin untuk menyelesaikan tantangan tersebut.

Program ini menghadirkan permainan yang menampilkan pertandingan antara Roro Jonggrang dan Bandung Bondowoso. Roro Jonggrang tidak ingin Bandung Bondowoso dapat menyelesaikan tantangan yang diberikan, sedangkan Bandung Bondowoso melakukan segala upaya agar dapat menyelesaikan tantangan yang diberikan supaya dia dapat menikahi Roro Jonggrang. Saat Program dijalankan, pemain diminta untuk login dan memilih karakter yang dimainkan. Jika pemain memilih Bandung Bondowoso, maka tujuan pemain dalam permainan ini adalah untuk membangun 100 candi, sedangkan jika pemain memilih Roro Jonggrang, maka tujuan pemain dalam permainan ini adalah untuk mencegah Bandung Bondowoso membangun 100 candi. Selain itu, jika pemain memilih peran sebagai Bandung Bondowoso, maka pemain juga dapat melakukan login sebagai jin yang membantu Bandung Bondowoso dalam pembangunan candi. Berikut adalah program yang dapat dilakukan oleh setiap karakter:

## 1. Bandung Bondowoso

- a. Summon Jin – Program ini berfungsi untuk memanggil jin baru. Jumlah maksimal jin yang dipanggil adalah 100 jin. Ada 2 jenis jin, yaitu jin pembangun dan jin pengumpul
  - > Jin Pembangun – Jin yang bertugas untuk membangun candi dari bahan yang sudah dikumpulkan sebelumnya
  - > Jin Pengumpul – Jin yang bertugas untuk mengumpulkan bahan yang digunakan untuk membangun candi
- b. Hilangkan Jin – Program ini berfungsi untuk menghapus jin yang sudah ada sebelumnya
- c. Ubah Tipe Jin – Program ini berfungsi untuk mengganti jenis jin
- d. Batch Kumpul / Bangun – Program ini berfungsi untuk mengarahkan jin untuk mengumpulkan bahan atau membangun candi
- e. Laporan Jin – Program ini berfungsi untuk mengetahui jumlah jin pengumpul, jin pembangun, jin terajin, jin termalas, total seluruh jin, serta bahan- bahan yang ada

- f. Laporan Candi – Program ini berfungsi untuk mengetahui total bahan yang digunakan, candi termahal, candi termurah, dan total candi yang ada
- 2. Roro Jonggrang
  - a. Hancurkan Candi – Program ini berfungsi untuk menghancurkan candi yang sudah dibangun oleh Bandung Bondowoso sebelumnya
  - b. Ayam Berkokok – Program ini berfungsi untuk menentukan pemenang dari game ini, apabila candi yang sudah dibuat oleh Bandung Bondowoso mencapai 100, maka Roro Jonggrang kalah, tetapi jika sebaliknya maka Roro Jonggrang menang
- 3. Jin
  - a. Bangun – Program ini hanya dapat dipanggil oleh jin pembangun, fungsi program ini adalah untuk membangun candi dari bahan- bahan yang sudah dikumpulkan sebelumnya
  - b. Kumpul – Program ini hanya dapat dipanggil oleh jin pengumpul, fungsi program ini adalah untuk mengumpulkan bahan- bahan seperti pasir, air, dan batu.

Pada permainan ini, terdapat juga beberapa program lain yang dapat dijalankan oleh semua karakter, yaitu:

- a. Login – Program ini berfungsi untuk mengetahui siapa pemain yang sedang menjalankan program
- b. Logout – Program ini berfungsi untuk mengganti karakter pemain
- c. Save – Program ini berfungsi untuk menyimpan data program- program yang sudah dijalan sebelumnya ke suatu file tertentu
- d. Load – Program ini berfungsi untuk membaca data dari suatu file tertentu
- e. Help – Untuk menampilkan bantuan yang berisi command yang dapat dipanggil oleh user sesuai dengan karakter yang sedang digunakan dalam program
- f. Exit – Program ini berfungsi untuk keluar dari permainan (menghentikan permainan)

# Daftar Pembagian Kerja Kelompok

Tabel 2.1- Tabel Daftar Pembagian Kerja Kelompok

Fitur	Implementasi	NIM Desainer	NIM Coder	NIM Tester
F01 - Login	function login	19622078	19622078	19622078
F02 - Logout	function logout	16522024 19622078	16522024 19622078	16522024
F03 - Summon Jin	procedure summonjin	19622024 19622078	19622024 19622078	19622024
F04 - Hilangkan Jin	procedure hapusjin	19622150	19622150 19622078	19622150
F05 - Ubah Tipe Jin	procedure ubahjin	19622078	19622078	19622078
F06 - Jin Pembangun	function bangun	16522024 19622078	16522024 19622078	16522024
F07 - Jin Pengumpul	function kumpul	19622024 19622078	19622024 19622078	19622024
F08 - Batch Kumpul / Bangun	procedure batchkumpul, procedure batchbangun	19622150	19622150 19622078	19622150
F09 - Ambil Laporan Jin	procedure laporanjin	19622078	19622078	19622078
F10 - Ambil Laporan Candi	procedure laporancandi	16522024 19622078	19622078	16522024
F11 - Hancurkan Candi	procedure hancurkancandi	19622024 19622078	19622078	19622024
F12 - Ayam Berkokok	procedure ayamberkokok	19622150	19622150 19622078	19622150
F13 - Load	function load	19622078	19622078	19622078
F14 - Save	procedure save	16522024	16522024	16522024



		19622078	19622078	
F15 - Help	procedure help	19622024	19622024 19622078	19622024
F16 - Exit	procedure exit	19622150	19622150	19622150
B01 - Random Number Generator	function randomize	19622150	19622150	19622150
B02 - Rekursif	function bangun, function kumpul, function string_split, function int_min	19622078	19622078	19622078
B03 - Typing	Semua function dan procedure	19622078	19622078	19622078
B04 - Undo	procedure undohapus	19622078	19622078	19622078
B0?? - Kreativitas	-	-	-	-

# Checklist Hasil Rancangan

Tabel 2.2 Tabel Checklist Hasil Rancangan

Fitur	Desain	Implementasi	Testing
F01 - Login	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F02 - Logout	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F03 - Summon Jin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F04 - Hilangkan Jin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F05 - Ubah Tipe Jin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F06 - Jin Pembangun	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F07 - Jin Pengumpul	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F08 - Batch Kumpul / Bangun	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F09 - Ambil Laporan Jin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F10 - Ambil Laporan Candi	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F11 - Hancurkan Candi	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F12 - Ayam Berkokok	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F13 - Load	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F14 - Save	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F15 - Help	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
F16 - Exit	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B01 - Random Number Generator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B02 - Rekursif	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B03 - Typing	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

B04 - Undo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B0?? - Kreativitas	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

# Desain Command

## F01 - Login

Initial State : User dalam keadaan belum login atau sudah logout dari akun sebelumnya

Final State : User telah melakukan login dengan username dan password yang sudah terdaftar sebelumnya dan mendapatkan role tertentu (role : bandung bondowoso, roro jonggrang, jin pembangun, jin pengumpul)

### – Simulasi Program –

#Kasus username tidak terdaftar

```
>>> login
Username :Banteng
Password :cintaroro
Username tidak terdaftar!
```

#Kasus password salah

```
>>> login
Username :Bondowoso
Password :gacintaroro
Password salah!
```

#Kasus username dan password valid

```
>>> login
Username :Bondowoso
Password :cintaroro
Selamat datang, Bondowoso!
Masukkan command "help" untuk daftar command yang dapat kamu panggil
```

#Kasus sudah melakukan login

```
>>> login
Login gagal!
Anda telah login dengan username Bondowoso, silahkan lakukan
"logout" sebelum melakukan login kembali.
```

## F02 - Logout

Initial State : User telah melakukan login dengan role tertentu sehingga terkategori sebagai user yang telah melakukan login

Final State : User keluar dari akun dan akan terkategori sebagai user yang belum melakukan login

### – Simulasi Program –

```
#Kasus user telah melakukan login
```

```
>>> logout
```

```
Logout berhasil!
```

```
#Kasus user belum melakukan login
```

```
>>> logout
```

```
Logout gagal!
```

```
Anda belum login, silahkan login terlebih dahulu sebelum melakukan  
logout
```

## F03 - Summon Jin

Initial State : User telah melakukan login dan memiliki role Bandung Bondowoso, jika user telah melakukan login tetapi tidak memiliki role sebagai bandung bondowoso, maka summon jin tidak dapat diakses

Final State : User menambahkan jin berdasarkan jenis jin yang telah dipilih, dan diberi username beserta password jin tersebut. Jika username sudah ada, maka akan muncul pesan kesalahan bahwa username jin telah diambil dan akan diminta username kembali, jika password yang dimasukkan kurang dari 5 karakter atau lebih dari 25 karakter, maka akan muncul pesan kesalahan bahwa password harus memiliki 5 - 25 karakter

### – Simulasi Program –

```
>>> summonjin
```

```
Jenis jin yang dapat dipanggil:
```

```
(1) Pengumpul - Bertugas mengumpulkan bahan bangunan
```

(2) Pembangun - Bertugas membangun candi

Masukkan nomor jenis jin yang ingin dipanggil: 3

Tidak ada jenis jin bernomor "3"!

Masukkan nomor jenis jin yang ingin dipanggil: 2

Memilih jin "Pembangun".

Masukkan username jin: jin1

Username "jin1" sudah diambil!

Masukkan username jin: asteron

Masukkan password jin: ast

Password panjangnya harus 5-25 karakter!

Masukkan password jin: asteronwinner

Mengumpulkan sesajen...

Menyerahkan sesajen...

Membacakan mantra...

Jin asteron berhasil dipanggil!

#### **F04 - Hilangkan jin**

Initial State : User telah melakukan login dengan role Bandung Bondowoso, jika login dengan role lain, maka hilangkan jin tidak dapat di akses. Memiliki setidaknya 1 jin yang sudah terdefinisi usernamenya

Final State : Jin yang dipilih dan terdefinisi usernamenya akan hilang / terhapus beserta dengan candi yang telah dibuatnya. Selain kondisi tersebut, program tidak akan melakukan apa-apa

### - Simulasi Program -

```
#Kasus username jin tidak ada
>>> hapusjin
Masukkan username jin : asterron
Tidak ada jin dengan username tersebut.
```

```
#Kasus username jin ada
>>> hapusjin
Masukkan username jin : asteron
Apakah anda yakin ingin menghapus jin dengan username Jin1 (Y/N)? Y

Jin telah berhasil dihapus dari alam gaib.
```

### F05 - Ubah Tipe Jin

Initial State : Telah melakukan login dengan role Bandung Bondowoso, apabila login dengan role lain, maka ubah tipe jin tidak dapat diakses. Memiliki setidaknya 1 jin dengan username dan password yang sudah terdefinisi untuk dapat diubah tipenya

Final State : Jin yang dipilih akan berubah tipenya dari jin pembangun menjadi jin pengumpul atau dari jin pengumpul menjadi jin pembangun

### - Simulasi Program -

```
#Kasus username jin tidak ada
>>> ubahjin
Masukkan username jin: kodir

Tidak ada jin dengan username tersebut.
```

```
#Kasus username jin ada
>>> ubahjin
Masukkan username jin: asteron
Jin ini bertipe "Pembangun". Yakin ingin mengubah ke tipe "Pengumpul" (Y/N)? Y
Jin telah berhasil diubah.
```

## F06 - Jin Pembangun

Initial State : Telah melakukan login dengan role jin pembangun, apabila login dengan role lain, jin pembangun tidak dapat diakses. Telah memiliki setidaknya 1 jin pembangun.

Final State : Akan terbangun candi jika bahan bangunan mencukupi, dan tercetak ke layar sisa candi yang perlu dibangun. Apabila bahan bangunan tidak mencukupi, akan tercetak ke layar bahwa bahan bangunan tidak mencukupi dan candi tidak dapat dibuat.

### – Simulasi Program –

```
>>> bangun
Candi berhasil dibangun.
Sisa candi yang perlu dibangun: 92
```

```
>>> bangun
# Men-generate bahan bangunan (4 pasir, 1 batu, dan 4 air)
# Memiliki 2 pasir, 2 batu, 6 air
Bahan bangunan tidak mencukupi.
Candi tidak bisa dibangun!
```

## F07 - Jin Pengumpul

Initial State : Telah melakukan login dengan role jin pengumpul, apabila login dengan role lain, fitur jin pengumpul tidak dapat diakses. Telah memiliki setidaknya 1 jin pengumpul yang sudah terdefinisi

Final State : Jin pengumpul akan mengumpulkan / mencari bahan bangunan untuk membuat candi (pasir, batu, dan air)

### – Simulasi Program –

```
>>> kumpul
Jin menemukan 1 pasir, 4 batu, dan 1 air.
```



```
>>> kumpul  
Jin menemukan 5 pasir, 5 batu, dan 2 air.
```

#### F08 - Batch Kumpul / Bangun

Initial State : Telah melakukan login dengan role Bandung Bondowoso, apabila login dengan role lain, fitur batch kumpul / bangun tidak dapat diakses. Memiliki setidaknya 1 jin untuk dapat dipanggil.

Final State : Akan dipanggil semua jin untuk membangun candi atau mengumpulkan bahan bangunan untuk membuat candi

#### – Simulasi Program Batch Kumpul –

```
>>> batchkumpul  
Mengerahkan 8 jin untuk mengumpulkan bahan.  
Jin menemukan total 20 pasir, 22 batu, dan 24 air.
```

#### – Simulasi Program Batch Bangun –

```
>>> batchbangun  
Mengerahkan 7 jin untuk membangun candi dengan total bahan 23 pasir,  
18 batu, dan 27 air.  
Jin berhasil membangun total 7 candi.
```

```
>>> batchbangun  
Mengerahkan 7 jin untuk membangun candi dengan total bahan 23 pasir,  
16 batu, dan 18 air.  
12 pasir Bangun gagal. Kurang 0 pasir.
```

#### F09 - Ambil Laporan Jin

Initial State : Telah melakukan login dengan role Bandung Bondowoso, apabila login dengan role lain, fitur ambil laporan jin tidak dapat diakses

Final State : Tercetak ke layar informasi tentang total jin, total jin pengumpul, total jin pembangun, jin terajin, jin termalas, dan jumlah bahan bangunan yang terkumpul (jumlah pasir, batu, dan air)

**- Simulasi Program -**

```
#Login : Jin Pembangun
>>> laporanjin
Laporan jin hanya dapat diakses oleh akun Bandung Bondowoso.
```

```
>>> laporanjin
> Total Jin: 15
> Total Jin Pengumpul: 8
> Total Jin Pembangun: 7
> Jin Terajin: asteron
> Jin Termalas: jin5
> Jumlah Pasir : 11 unit
> Jumlah Batu : 29 unit
> Jumlah Air : 17 unit
```

**F10 - Ambil Laporan Candi**

Initial State : Telah melakukan login dengan role Bandung Bondowoso, apabila login dengan role lain, fitur ambil laporan candi tidak dapat diakses

Final State : Tercetak ke layar informasi tentang total candi, total jumlah bahan bangunan yang terpakai / digunakan (jumlah pasir, batu, dan air), ID candi termahal, dan ID candi termurah

**- Simulasi Program -**

```
#Kasus login dengan role lain (Roro Jonggrang, Jin Pembangun, Jin
#Pengumpul)
>>> laporancandi
Laporan candi hanya dapat diakses oleh akun Bandung Bondowoso.
```

```
>>> laporancandi
> Total Candi: 32
> Total Pasir yang digunakan: 102
> Total Batu yang digunakan: 86
> Total Air yang digunakan: 93
> ID Candi Termahal: 1 (Rp 147500)
> ID Candi Termurah: 18 (Rp 40000)
```

## F11 - Hancurkan Candi

Initial State : Telah melakukan login dengan role Roro Jonggrang, apabila login dengan menggunakan role lain, fitur hancurkan candi tidak dapat diakses. Memiliki setidaknya 1 candi dengan ID yang terdefinisi untuk dapat dihancurkan

Final State : ID Candi yang dipilih akan hilang / terhapus dari daftar candi yang telah dibangun

### – Simulasi Program –

```
#Kasus login dengan role lain (Bandung Bondowoso, Jin Pembangun, Jin  
#Pengumpul)
```

```
>>> hancurkancandi
```

Hancurkan candi hanya dapat diakses oleh akun Roro Jonggrang.

```
>>> hancurkancandi
```

```
Masukkan ID candi: 980
```

Tidak ada candi dengan ID tersebut.

```
>>> hancurkancandi
```

```
Masukkan ID candi: 19
```

```
Apakah anda yakin ingin menghancurkan candi ID: 19 (Y/N)? Y
```

Candi telah berhasil dihancurkan.

## F12 - Ayam Berkokok

Initial State : Telah melakukan login dengan role Roro Jonggrang, apabila login dengan role lain, fitur ayam berkokok tidak dapat diakses / dijalankan

Final State : Jika terdapat 100 candi yang telah dibuat, maka tercetak pesan ke layar bahwa Bandung Bondowoso memenangkan permainan, sedangkan jika terdapat kurang dari 100 candi yang telah dibuat, tercetak pesan ke layar bahwa Roro Jonggrang memenangkan permainan dan dikutuk menjadi candi oleh Bandung Bondowoso. Setelah tercetak salah satu pesan tersebut, program akan keluar secara otomatis

### – Simulasi Program –

```
>>> ayamberkokok
Kukuruyuk.. Kukuruyuk..
Jumlah Candi: 32

Selamat, Roro Jonggrang memenangkan permainan!

*Bandung Bondowoso angry noise*
Roro Jonggrang dikutuk menjadi candi.
```

```
>>> ayamberkokok
Kukuruyuk.. Kukuruyuk..
Jumlah Candi: 100
Yah, Bandung Bondowoso memenangkan permainan!
```

### F13 - Load

Initial State : Program utama belum berjalan dan folder data sudah ada pada parent folder "save"

Final State : Program sudah berjalan dengan folder data sudah terakses dan tersimpan dalam program, yaitu dalam variabel users, candi, dan bahan\_bangunan.

### – Simulasi Program –

```
#Folder yang akan diloat tidak ditemukan / tidak ada
~$ python main.py tester_kel11
Folder "save/tester_kel11" tidak ditemukan.
```

```
#Tidak memberikan nama folder yang akan diloat
~$ python main.py
Tidak ada nama folder yang diberikan!

usage: main.py <nama_folder>
```

```
#Folder yang akan diloat ditemukan / ada
~$ python main.py tester_kel10
Loading...
Selamat datang di program "Manajerial Candi"!
Masukkan command "help" untuk daftar command yang dapat kamu panggil.
```

#### F14 - Save

Initial State : Telah melakukan load data dan masuk ke program manajerial candi

Final State : Menyimpan data hasil permainan ke dalam sebuah folder yang telah ditentukan namanya di dalam parent folder "save", jika folder tidak ada maka dibuat oleh program

#### – Simulasi Program –

```
#Save pada folder baru
>>> save
Masukkan nama folder: tester_kel10
Saving...
Membuat folder save/tester_kel10...

Berhasil menyimpan data di folder save/tester_kel10!
```

```
#Save pada folder yang sudah ada
>>> save
Masukkan nama folder: tester_kel10
Saving...
Berhasil menyimpan data di folder save/tester_kel10!
```

#### F15 - Help

Initial State : Telah melakukan load data dan masuk ke program utama

Final State : Mencetak pesan ke layar berupa beberapa perintah / program yang bisa dijalankan oleh user. Pesan ini akan berbeda-beda tergantung dari status login atau status role yang dipilih user

#### – Simulasi Program –

```
#Belum melakukan login
>>> help
===== HELP =====
1. login
   Untuk masuk menggunakan akun
```

2. save  
Untuk menyimpan data permainan saat ini
3. exit  
Untuk keluar dari program dan kembali ke terminal

#Login : Bandung Bondowoso

>>> help

===== HELP =====

1. logout  
Untuk keluar dari akun yang digunakan sekarang
2. summonjin  
Untuk memanggil jin
3. hapusjin  
Untuk menghapus jin
4. undohapus  
Untuk mengembalikan jin yang terhapus
4. ubahjin  
Untuk mengubah tipe jin (pengumpul/pembangun)
5. batchkumpul  
Untuk melakukan pengumpulan oleh semua jin pengumpul
6. batchbangun  
Untuk melakukan pembangunan oleh semua jin pembangun
7. laporanjin  
Untuk mengambil laporan jin
8. laporancandi  
Untuk mengambil laporan candi
9. save  
Untuk menyimpan data permainan saat ini
10. exit  
Untuk keluar dari program dan kembali ke terminal

#Login : Roro Jonggrang

>>> help

===== HELP =====

1. logout  
Untuk keluar dari akun yang digunakan sekarang
2. hancurkancandi  
Untuk menghancurkan candi yang tersedia
3. ayamberkokok  
Untuk menyelesaikan permainan dengan memalsukan pagi hari
4. save  
Untuk menyimpan data permainan saat ini
5. exit  
Untuk keluar dari program dan kembali ke terminal

```
#Login : Jin Pembangun
>>> help
===== HELP =====
1. logout
   Untuk keluar dari akun yang digunakan sekarang
2. bangun
   Untuk membangun sebuah candi
3. save
   Untuk menyimpan data permainan saat ini
4. exit
   Untuk keluar dari program dan kembali ke terminal
```

```
#Login : Jin Pengumpul
>>> help
===== HELP =====
1. logout
   Untuk keluar dari akun yang digunakan sekarang
2. kumpul
   Untuk mengumpulkan bahan bangunan
3. save
   Untuk menyimpan data permainan saat ini
4. exit
   Untuk keluar dari program dan kembali ke terminal
```

#### F16 - Exit

Initial State : Program utama sudah berjalan dengan file / folder yang sudah terload

Final State : User akan diberikan opsi untuk menyimpan data (Prosedur Save (F14)) atau tidak, jika memilih tidak, maka program akan langsung keluar, jika memilih iya, program akan melakukan prosedur save (F14)

#### – Simulasi Program –

```
>>> exit
Apakah Anda mau melakukan penyimpanan file yang sudah diubah? (y/n) n
#Keluar Program
```

```
>>> exit
```

```
Apakah Anda mau melakukan penyimpanan file yang sudah diubah? (y/n) y
#Melakukan Prosedur save (F14)
Masukkan nama folder: tester_kel10
Saving...
Berhasil menyimpan data di folder save/tester_kel10!
#Keluar Program
```



# Desain Kamus Data

## Main Program

### **KAMUS**

```
type Data : < isi : array [0..n_baris-1] of
              array [0..n_kolom-1] of string,
              n_baris : integer,
              n_kolom : integer >
masukkan : string
semua_data : array[0..2] of Data
users, candi, bahan_bangunan : Data
```

## Modul command

### **KAMUS**

```
type Data : < isi : array [0..n_baris-1] of
              array [0..n_kolom-1] of string,
              n_baris : integer,
              n_kolom : integer >
const undo_maks : integer = 100
undo_jin : array [0..(2*undo_maks)-1] of
           array [0..3] of string
undo_candi : array [0..(10*undo_maks)-1] of
            array [0..3] of string
current_login : array [0..2] of string
```

### **procedure run**

```
(input command: string,
 input/output users: Data,
 input/output candi: Data,
 input/output bahan_bangunan : Data)
```

### **KAMUS LOKAL**

```
length : integer
command_list : array [0..length-1] of string
```

### **function login**

```
(current_login: array [0..2] of string,
 users: Data ) -> array [0..2] of string
```

**KAMUS LOKAL**

```
index : integer  
username, password : string  
isi_users : array[0..users.n_baris-1] of  
            array[0..users.n_kolom-1] of string
```

**function logout**

```
(current_login: array [0..2] of string)  
-> array [0..2] of string
```

**KAMUS LOKAL**

```
current_login : array [0..2] of string
```

**procedure summonjin**

```
(input/output users: Data)
```

**KAMUS LOKAL**

```
n_baris, i : integer  
jenis_jin, username, password, role : string  
username_valid : boolean  
isi_users : array [0..users.n_baris-1] of  
            array [0..users.n_kolom-1] of string
```

**procedure hapusjin**

```
(input/output users: Data,  
 input/output candi: Data,  
 input/output undo_jin: array[0..2*undo_maks] of  
                        array[0..3] of string,  
 input/output undo_candi: array[0..10*undo_maks] of  
                        array[0..3] of string )
```

**KAMUS LOKAL**

```
n_baris_candi, count_candi_hapus, i, j, index : integer  
username, jawab : string  
candi_dihapuskan : < isi : array[0..length-1] of integer,  
                    length: integer >  
isi_candi : array [0..candi.n_baris-1] of  
            array [0..candi.n_kolom-1] of string
```

**procedure undohapus**

```
(input/output users: Data,  
 input/output candi: Data,  
 input/output undo_jin: array [0..(2*undo_maks)-1] of  
                        array [0..3] of string,  
 input/output undo_candi: array [0..(10*undo_maks)-1] of  
                        array [0..3] of string )
```

**KAMUS LOKAL**

```
i, j, count_undo_candi : integer  
username : string
```

### **procedure ubahjin**

```
( input/output users: Data )
```

#### **KAMUS LOKAL**

```
n_baris, i, index : integer  
username, ubah: string  
ditemukan : boolean  
isi_users : array [0..users.n_baris-1] of  
array [0..users.n_kolom-1] of string
```

### **function bangun**

```
(candi : Data,  
bahan_bangunan : Data,  
jin_pembangun : array [0..length-1] of string,  
length : integer,  
jumlah_loop : integer,  
bangun_batch : boolean )  
-> array [0..length-1] of array [0..3] of string
```

#### **KAMUS LOKAL**

```
n_baris_candi : integer  
bahan_dimiliki, bahan_diperlukan : array[0..2] of integer  
data_hasil : array[0..3] of string  
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of  
array [0..bahan_bangunan.n_kolom-1] of  
string
```

### **function kumpul**

```
(bahan_bangunan : Data,  
jumlah_loop : integer,  
kumpul_batch : boolean )-> array [0..2] of string
```

#### **KAMUS LOKAL**

```
bahan_dikumpul : array[0..2] of integer  
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of  
array [0..bahan_bangunan.n_kolom-1] of  
string
```

### **procedure batchkumpul**

```
(input users: Data,  
input/output bahan_bangunan: Data )
```

#### **KAMUS LOKAL**

```
i, n_baris_users, count_jin_pengumpul : integer  
hasil_kumpul : array[0..2] of integer
```

```
isi_users : array [0..users .n_baris-1] of  
            array [0..users.n_kolom-1] of string
```

**procedure batchbangun**

```
(input users: Data,  
  input/output candi: Data,  
  input/output bahan_bangunan: Data )
```

**KAMUS LOKAL**

```
i, index, n_baris_users, count_jin_pembangun, total_pasir,  
total_batu, total_air, kurang_pasir, kurang_batu,  
kurang_air : integer
```

```
bahan_dimiliki : array[0..2] of integer  
jin_pembangun : array[0..count_jin_pembangun-1] of string  
isi_users : array [0..users.n_baris-1] of  
            array [0..users.n_kolom-1] of string  
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of  
            array [0..bahan_bangunan.n_kolom-1] of  
                string  
data_pembangunan : array [0..count_jin_pembangun-1] of  
            array [0..3] of string
```

**procedure laporanjin**

```
(input users: Data,  
  input candi: Data,  
  input bahan_bangunan: Data)
```

**KAMUS LOKAL**

```
n_baris_users, n_baris_candi, total_jin, total_pengumpul,  
total_pembangun, i, index, n_jin, count_maks, count_min,  
candi_maks candi_min, index_jin_maks,  
index_jin_min : integer
```

```
count_candi_jin : array [0..n_jin-1] of integer  
jin_maks : array [0..count_maks-1] of string  
jin_min : array [0..count_min-1] of string  
jin : array [0..total_pembangun-1] of string  
isi_users : array [0..users.n_baris-1] of  
            array [0..users.n_kolom-1] of string  
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of  
            array [0..bahan_bangunan.n_kolom-1] of  
                string  
data_pembangunan : array [0..count_jin_pembangun-1] of  
            array [0..3] of string
```

**procedure laporan candi**

(input candi: Data)

**KAMUS LOKAL**

i, n\_baris\_candi, harga\_maks, harga\_min, total\_pasir,  
total\_batu, total\_air : integer

harga : array [0..n\_baris\_candi-1] of integer  
isi\_candi : array [0..candi.n\_baris-1] of  
array [0..candi.n\_kolom-1] of string

**procedure hancurkancandi**

(input/output candi: Data)

**KAMUS LOKAL**

i, index, n\_baris\_candi : integer  
id, yakin : string  
isi\_candi : array [0..candi.n\_baris-1] of  
array [0..candi.n\_kolom-1] of string

**procedure ayamberkokok**

(input candi: Data)

**KAMUS LOKAL**

i, banyak\_candi : integer

**function load**

() -> array [0..2] of Data

**KAMUS LOKAL**

parser, args : objek dari modul argparse  
args.nama\_folder : string  
users, candi, bahan\_bangunan : Data

**procedure save**

(input users: Data,  
input candi: Data,  
input bahan\_bangunan: Data)

**KAMUS LOKAL**

countSplitted, i : integer  
path, current\_path : string  
file\_baru : boolean  
pathSplitted : array [0..countSplitted-1] of string

**procedure help**

(input current\_login: array [0..2] of string)

**KAMUS LOKAL**

```
current_login : array[0..2] of string
```

**procedure exit**

```
(input users: Data,  
input candi: Data,  
input bahan_bangunan: Data )
```

**KAMUS LOKAL**

```
jawab_exit : string
```

## **Modul numgen**

**KAMUS**

```
seed : integer
```

**function randomize**

```
(lower_bound : integer,  
upper_bound : integer ) -> integer
```

**KAMUS LOKAL**

```
const a : integer = 1583458089  
const b : integer = 1132489760  
m : integer
```

## **Modul tools**

**KAMUS**

```
type Data : < isi : array [0..n_baris-1] of  
array [0..n_kolom-1] of string,  
n_baris : integer,  
n_kolom : integer >
```

**function string\_split**

```
(stringx : string, key : string)  
-> array [0..count-1] of string
```

**KAMUS LOKAL**

```
count, i, index_str, index_hasil : integer  
part : string  
hasil : array [count-1] of string
```

**function string\_slice**

```
(stringx : string,  
index_awal : integer,  
index_akhir : integer,  
index_sekarang : integer = 0)  
-> string
```

**KAMUS LOKAL**

```
index_sekarang : integer
```

**function string\_strip**

```
(stringx : string)  
-> array [index_awal..index_akhir-1] of string
```

**KAMUS LOKAL**

```
index_awal, index_akhir, i : integer  
hasil : string
```

**function string\_append**

```
(array : array [0..length-1] of string,  
stringx: string, length : integer)  
-> array [0..length] of string
```

**KAMUS LOKAL**

```
i : integer  
array_baru : array [0..length] of string
```

**function string\_in\_array**

```
(array : array [0..array_length-1] of string,  
stringx : string, array_length : integer) -> integer
```

**KAMUS LOKAL**

```
i : integer
```

**function string\_leksikografis\_min**

```
(array : array [0..length-1] of string,  
length : integer) -> string
```

**KAMUS LOKAL**

```
i : integer  
min : string
```

**function string\_leksikografis\_maks**

```
(array : array [0..length-1] of string,  
length : integer) -> string
```

**KAMUS LOKAL**

```
i : integer  
maks : string
```

**function int\_min**

```
(array : array [0..length-1] of integer, length : integer,  
index_sekarang : integer = 0, min : integer = 999999 )  
-> integer
```

**KAMUS LOKAL**

```
min, length : integer
```

**function int\_maks**

```
(array : array [0..length-1] of integer, length : integer)  
-> integer
```

**KAMUS LOKAL**

```
maks, i : integer
```

**function int\_join**

```
(array1 : array [0..length-1] of integer, array2 : array  
[0..length-1] of integer, length : integer)  
-> integer
```

**KAMUS LOKAL**

```
i : integer  
array_hasil : array [0..length-1] of integer
```

**function data\_append**

```
(data : Data, elemen: array[0..data.n_kolom-1] of string)
```

**KAMUS LOKAL**

```
n_baris, i : integer  
hasil : array [0..n_baris) of string  
isi_data : array [0..data.n_baris-1] of  
array [0..data.n_kolom-1] of string
```

**function data\_remove****KAMUS LOKAL**

```
n_baris, n_kolom, i : integer  
data_baru : array [0..n_baris-2] of  
array [0..n_kolom-1] of string  
isi_data : array [0..n_baris-1] of  
array [0..n_kolom-1] of string
```

**function cari\_index\_username****KAMUS LOKAL**

```
n_baris_users, i : integer  
isi_data : array [0..users.n_baris-1] of
```



```
array [0..users.n_kolom-1] of string
```

#### **function cari\_index\_candi**

##### **KAMUS LOKAL**

```
n_baris_candi, i, count_candi : integer  
indexs : array [0..count_candi-1] of integer  
isi_candi : array [0..candi.n_baris-1] of  
array [0..candi.n_kolom-1] of string
```

#### **function matrix\_str\_join**

##### **KAMUS LOKAL**

```
i : integer  
hasil : array [0..n_baris] of  
array [0..n_kolom-1] of string
```

#### **function candi\_append**

##### **KAMUS LOKAL**

```
n_baris_candi, n_kolom_candi, i, index: integer  
isi_candi : array [0..candi.n_baris-1] of  
array [0..candi.n_kolom-1] of string  
isi_candi_baru : array [0..candi.n_baris] of  
array [0..candi.n_kolom-1] of string
```

#### **function generate\_bahan\_bangunan**

##### **KAMUS LOKAL**

```
pasir, batu, air: integer
```

## **Modul data**

##### **KAMUS**

```
type Data : < isi : array [0..n_baris-1] of  
array [0..n_kolom-1] of string,  
n_baris : integer,  
n_kolom : integer >
```

#### **function data\_load**

```
(path : string) -> Data
```

##### **KAMUS LOKAL**

```
n_baris, n_kolom, i : integer  
line : string  
cek_kolom, first_row : boolean
```

```
file : open(path, "r")  
data : Data
```

**function data\_save**

```
(input path : string,  
input nama_file : string,  
input data : Data)
```

**KAMUS LOKAL**

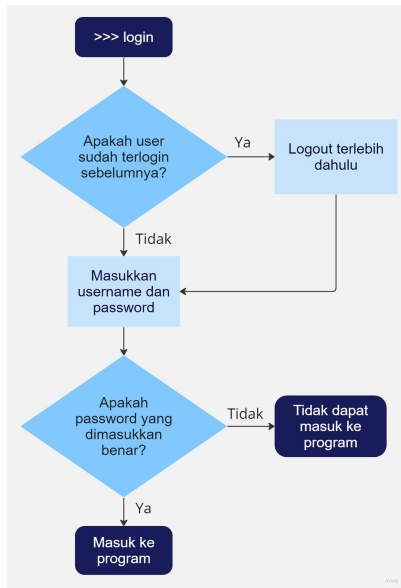
```
n_baris, n_kolom, i, j : integer  
line : string  
file : open(path, "w+")  
isi_data : matriks of string
```

# Desain Dekomposisi Algoritmik dan Fungsional

## Program

### 1. F01 - Login

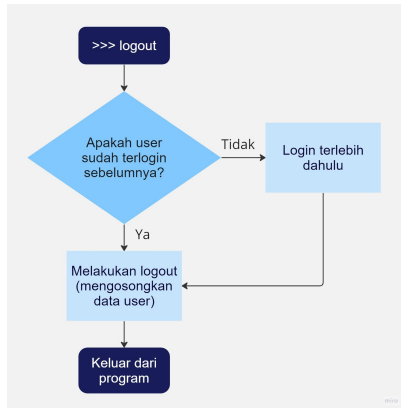
Fungsi ini adalah langkah awal user untuk masuk ke program dan memulai permainan. Pada fungsi ini, user diharuskan untuk memasukkan username dan password yang sudah terdaftar sebelumnya. Jika username dan password benar, maka pengguna akan masuk ke dalam program. Namun, jika pengguna salah memasukkan username atau password, maka pengguna tidak akan masuk ke dalam program. Jika user sudah pernah login sebelumnya, maka user harus melakukan logout terlebih dahulu sebelum melakukan login kembali.



Flowchart Fungsi F01

### 2. F02 - Logout

Fungsi ini merupakan langkah user untuk keluar dari program. Fungsi ini hanya dapat dijalankan apabila pemain sudah melakukan login sebelumnya. Logout dilakukan dengan cara mengosongkan data user yang sudah terlogin.

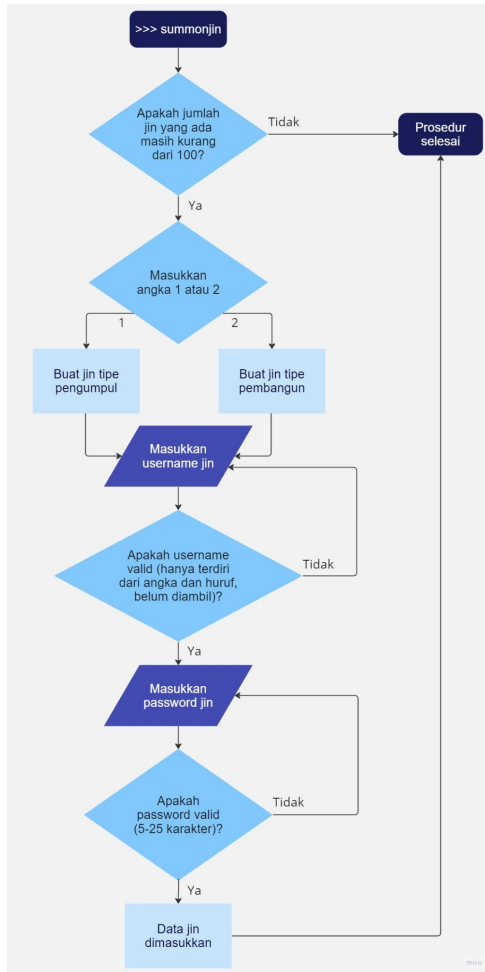


Flowchart Fungsi F02

### 3. F03 - Summon Jin

Fungsi ini dapat membuat jin baru. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso. Jumlah maksimal jin yang dapat dibuat menggunakan fungsi ini adalah 100 jin. Fungsi tidak dapat dijalankan lagi ketika jumlah jin telah mencapai batas ini. Selama fungsi masih dapat dijalankan, user dapat memilih antara dua jenis jin, yaitu jin pengumpul, yang dapat dipanggil dengan memasukkan angka 1, dan jin pembangun, yang dapat dipanggil dengan memasukkan angka 2. Apabila user memasukkan angka selain 1 dan 2, prosedur permintaan input akan diulang hingga user memasukkan angka yang benar.

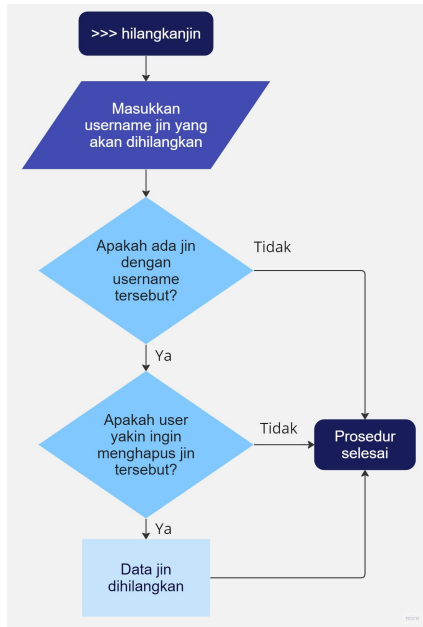
Setelah jin dipanggil, user akan diminta untuk memberikan username dan password untuk jin tersebut. Username dan password memiliki ketentuan yang harus diikuti. Username hanya boleh terdiri dari angka dan huruf, dan user tidak boleh menggunakan username yang sudah diambil sebelumnya. Adapun password harus memiliki panjang di antara 5–25 karakter. Apabila username dan password tidak valid, prosedur permintaan input akan diulang hingga user membuat username dan password yang valid. Data jin akan ditambahkan ke dalam database jika username dan password sudah valid.



Flowchart Fungsi F03

#### 4. F04 - Hilangkan Jin

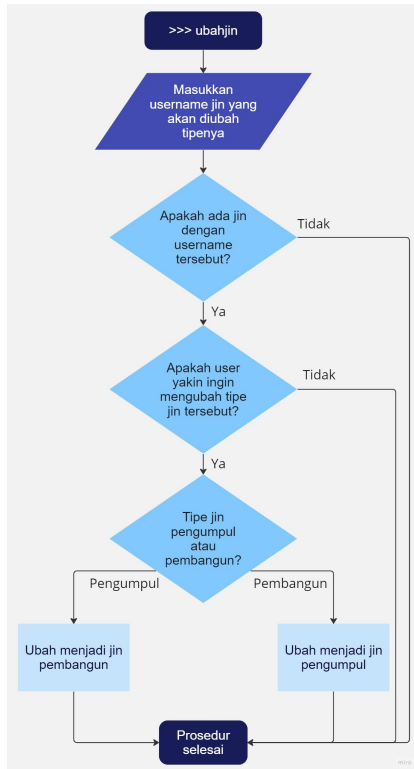
Fungsi ini dapat menghapus jin yang sudah ada. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso. Penghapusan jin dilakukan dengan cara memasukkan username jin yang akan dihapus. Sebelum jin dihapus, user harus melakukan konfirmasi terlebih dahulu. User diberikan pilihan untuk melakukan input berupa "Y" (yes) atau "N" (no). Jin hanya akan dihapus apabila user memasukkan input "Y".



Flowchart Fungsi F04

##### 5. F05 - Ubah Tipe Jin

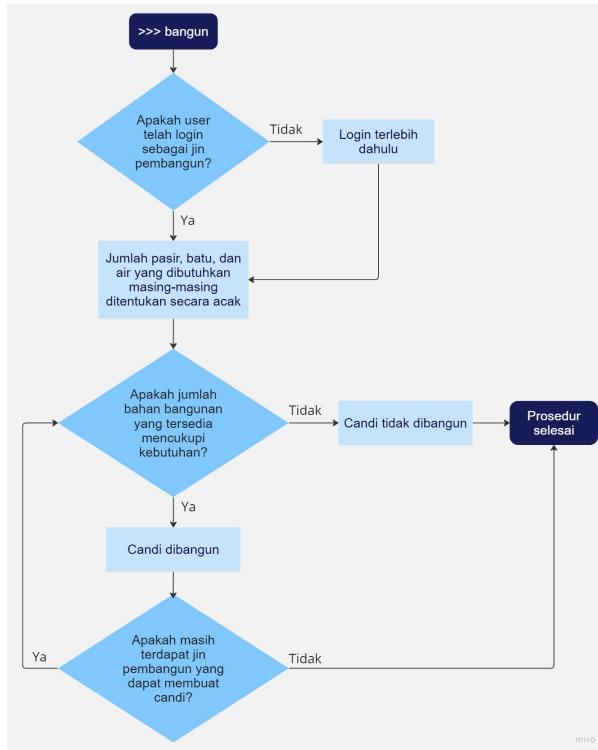
Fungsi ini dapat mengubah tipe jin yang sudah ada. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso. User dapat mengubah jin pengumpul menjadi pembangun, atau sebaliknya. Fungsi ini dijalankan dengan cara memasukkan username jin yang akan diubah tipenya. Program akan mencari username jin yang telah terdaftar dan mendapatkan informasi tipe jin dengan username tersebut. Fungsi tidak akan dijalankan jika username tidak dapat ditemukan. Jin dengan tipe pengumpul akan diubah menjadi tipe pembangun, dan jin dengan tipe pembangun akan diubah menjadi tipe pengumpul. Sebelum mengubah tipe jin, user harus melakukan konfirmasi terlebih dahulu. User diberikan pilihan untuk melakukan input berupa "Y" (yes) atau "N" (no). Jin hanya akan diubah tipenya apabila user memasukkan input "Y".



Flowchart Fungsi F05

#### 6. F06 - Jin Pembangun

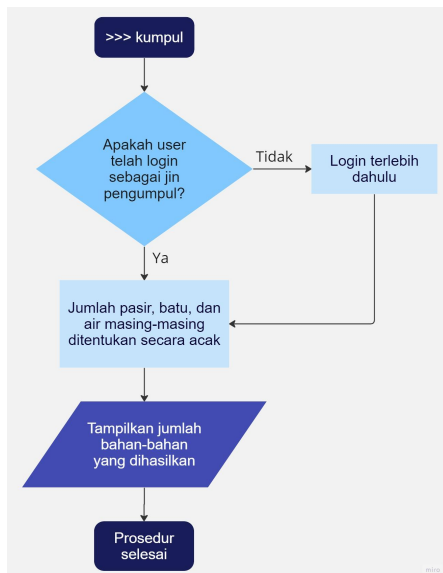
Fungsi ini dibuat secara khusus untuk jin tipe pembangun, yang bertugas membangun candi menggunakan bahan bangunan yang ada, berupa pasir, batu, dan air. Ketika fungsi ini dijalankan, user dapat membangun candi dengan jumlah bahan bangunan yang acak. Candi hanya akan dibangun apabila bahan yang tersedia cukup. Jumlah bahan bangunan yang dibutuhkan akan bersifat acak dan ditentukan oleh program itu sendiri. Pembangunan dilakukan secara rekursif apabila terdapat lebih dari satu jin pembangun.



Flowchart Fungsi F06

## 7. F07 - Jin Pengumpul

Fungsi ini dibuat secara khusus untuk jin tipe pengumpul, yang bertugas mengumpulkan bahan untuk membangun candi. Ketika fungsi ini dijalankan, user dapat mengumpulkan bahan bangunan berupa pasir, batu, dan air dalam jumlah acak. Program menghasilkan masing-masing bahan tersebut secara acak, kemudian menampilkan jumlahnya.

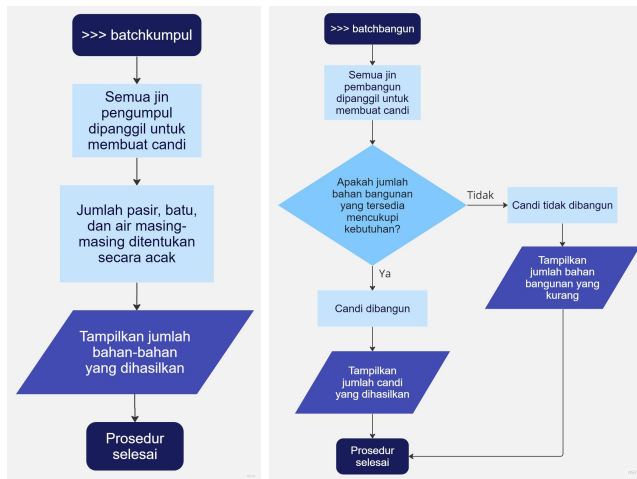


Flowchart Fungsi F07



#### 8. F08 - Batch Kumpul/Bangun

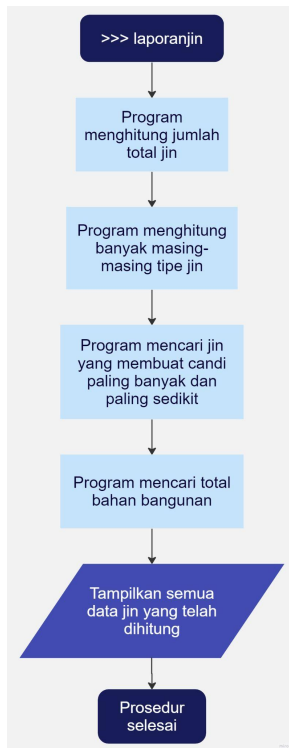
Fungsi ini digunakan untuk memanggil semua jin dari tipe tertentu untuk menjalankan tugasnya. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso. Prosedur batch kumpul digunakan untuk memanggil jin pengumpul, sedangkan batch bangun digunakan untuk memanggil jin pembangun. Untuk batch bangun, jumlah candi yang dibangun bergantung pada jumlah bahan bangunan yang tersedia.



Flowchart Fungsi F08 (batchkumpul dan batchbangun)

#### 9. F09 - Ambil Laporan Jin

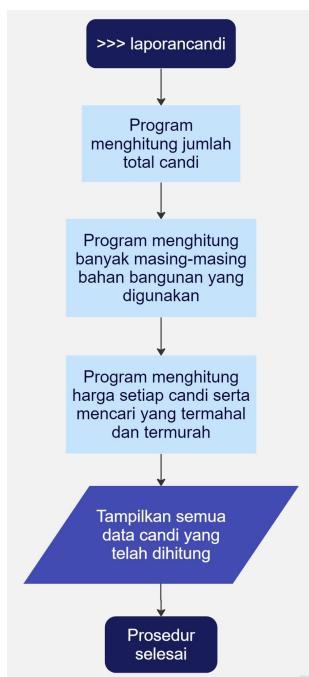
Fungsi ini digunakan untuk mengumpulkan data berupa jumlah total jin dari masing-masing tipe, jin “termalas”, jin “terajin”, serta jumlah bahan bangunan. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso. Ketika fungsi ini digunakan, data yang dibutuhkan akan diambil. Total jin dihitung dengan melakukan traversal melalui database user yang berisi setiap jin serta tipenya masing-masing. Kemudian, jin “termalas” adalah jin yang membangun candi yang paling sedikit, sebaliknya jin “terajin” adalah jin yang membangun candi yang paling banyak.



Flowchart Fungsi F09

#### 10. F10 - Ambil Laporan Candi

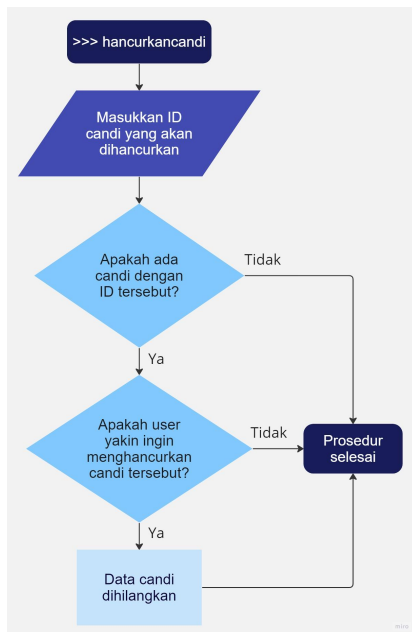
Fungsi ini digunakan untuk mengumpulkan data berupa jumlah candi, bahan total yang diperlukan, serta candi termurah dan termahal. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso.



Flowchart Fungsi F10

### 11. F11 - Hancurkan Candi

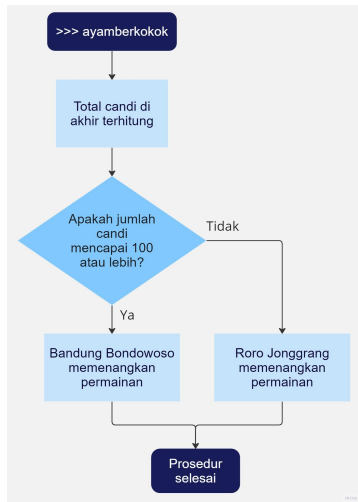
Fungsi ini dapat menghapus candi yang sudah ada. Fungsi ini dapat diakses ketika user berperan sebagai Bandung Bondowoso. Penghapusan candi dilakukan dengan cara memasukkan ID candi yang akan dihancurkan. ID candi ini berupa integer. Apabila username tidak dapat ditemukan, prosedur permintaan input akan diulang hingga user memasukkan ID yang ada dalam database. Sebelum candi dihancurkan, user harus melakukan konfirmasi terlebih dahulu. User diberikan pilihan untuk melakukan input berupa "Y" (yes) atau "N" (no). Candi hanya akan dihancurkan apabila user memasukkan input "Y".



Flowchart Fungsi F11

### 12. F12 - Ayam Berkokok

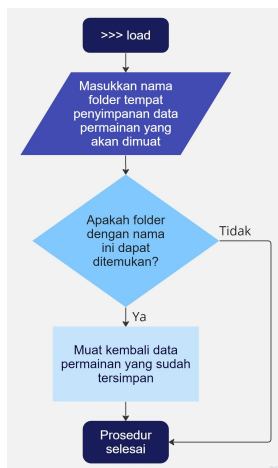
Fungsi ini mengakhiri sesi permainan yang sedang berjalan dan menghitung jumlah candi yang pada akhirnya dibangun untuk menentukan pemenang. Fungsi ini dapat diakses ketika user berperan sebagai Roro Jonggrang. Jika jumlah candi yang terhitung mencapai 100 atau lebih, fungsi akan memberikan keluaran bahwa Bandung Bondowoso memenangkan permainan. Sedangkan apabila jumlah candi kurang dari itu, Roro Jonggrang memenangkan permainan.



Flowchart Fungsi F12

### 13. F13 - Load

Fungsi ini memuat kembali data permainan yang sebelumnya sudah disimpan. Fungsi ini menggunakan `argparse` untuk menyimpan dan memuat data tersebut. User diminta untuk memasukkan nama folder yang berisi data permainan yang akan diakses. Jika folder yang dimuat tidak ada, fungsi tidak akan berjalan karena tidak ada yang dapat dimuat.



Flowchart Fungsi F13

### 14. F14 - Save

Fungsi ini menyimpan data permainan pada folder dengan *parent folder* "save". Ketika fungsi dijalankan, user dapat memasukkan nama folder penyimpanan. Apabila nama folder yang dimasukkan sudah ada dalam *parent folder*, data dapat langsung disimpan dalam folder tersebut. Apabila nama folder belum ada, folder baru dengan nama tersebut akan dibuat dan data disimpan dalam folder tersebut.



Flowchart Fungsi F14

### 15. F15 - Help

Fungsi ini menampilkan daftar berisi setiap perintah yang dapat digunakan oleh user sesuai dengan perannya. Fungsi akan memeriksa status login pemain terlebih dahulu, kemudian memberikan panduan yang sesuai.

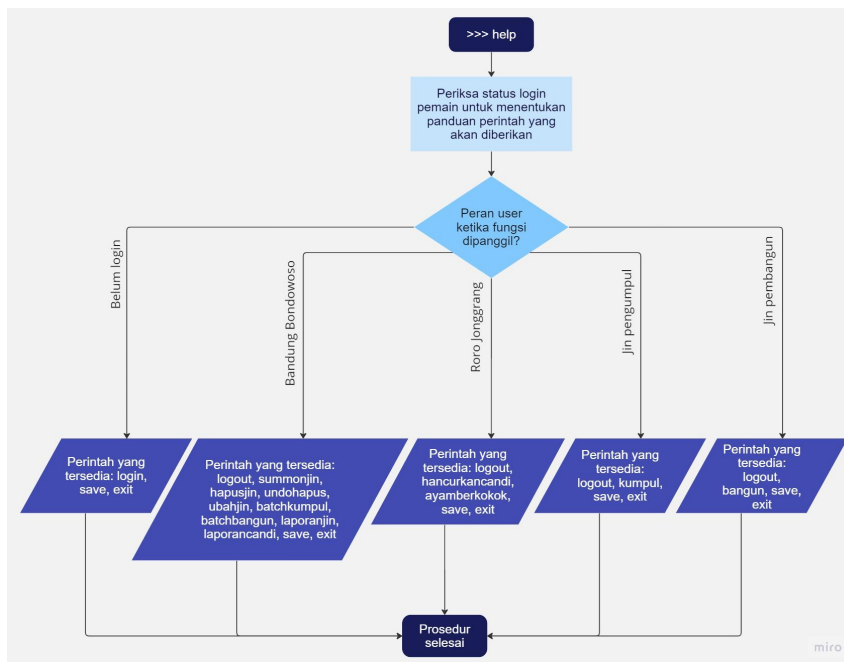
Perintah untuk user yang belum melakukan login: login, save, exit

Perintah untuk Bandung Bondowoso: logout, summonjin, hapusjin, undohapus, ubahjin, batchkumpul, batchbangun, laporanjin, laporancandi, save, exit

Perintah untuk Roro Jonggrang: logout, hancurkancandi, ayamberkokok, save, exit

Perintah untuk jin pengumpul: logout, kumpul, save, exit

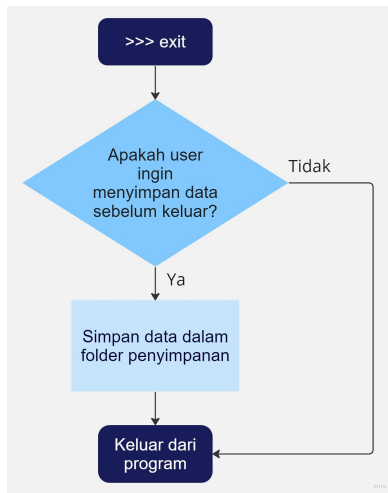
Perintah untuk jin pembangun: logout, bangun, save, exit



Flowchart Fungsi F15

#### 16. F16 - Exit

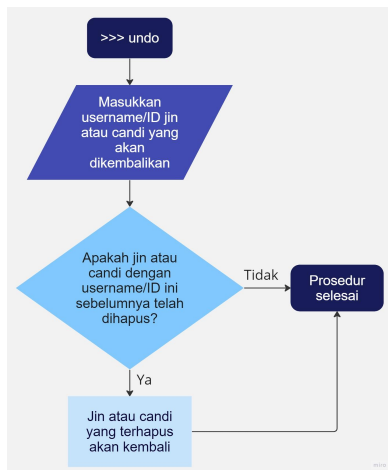
Fungsi ini digunakan user untuk keluar dari program. Sebelum fungsi dijalankan, user diberikan opsi untuk menyimpan data terlebih dahulu. Data akan disimpan di folder save apabila user memilih opsi tersebut. Jika user memilih untuk tidak menyimpannya, user akan langsung keluar dengan data yang tidak tersimpan.



Flowchart Fungsi F16

#### 17. B04 - Undo

Prosedur penghapusan jin (F04) maupun candi (F11) dapat dibatalkan. Fungsi ini dibuat untuk tujuan tersebut. Setelah penghapusan jin atau candi dibatalkan, keadaannya akan dikembalikan seperti semula.



Flowchart Fungsi B04

# Spesifikasi Program, Modul, dan Fungsi dalam Notasi Algoritmik

## Program Utama

### Program main

```
{ Program utama Manajerial Candi}
```

### KAMUS

```
type Data : < isi : array [0..n_baris-1] of  
            array [0..n_kolom-1] of string,  
            n_baris : integer,  
            n_kolom : integer >  
masukkan : string  
semua_data : array[0..2] of Data  
users, candi, bahan_bangunan : Data
```

### ALGORITMA

```
{import module commands yang berisi semua fungsi dan prosedur  
utama}  
import commands  
semua_data <- commands.load() { load data }  
if (semua_data ≠ []) then  
    { load berhasil }  
    users <- semua_data[0]  
    candi <- semua_data[1]  
    bahan_bangunan <- semua_data[2]  
    while True do  
        input(masukan)  
        commands.run(masukan,users,candi,bahan_bangunan)  
        output()
```

## Modul Commands

### Module commands

```
{ Berisi semua fungsi / prosedur untuk command-command yang dapat  
digunakan }
```

**KAMUS**

```
type Data : < isi : array [0..n_baris-1] of  
              array [0..n_kolom-1] of string,  
              n_baris : integer,  
              n_kolom : integer >  
const undo_maks : integer = 100  
undo_jin : array [0..(2*undo_maks)-1] of  
            array [0..3] of string  
undo_candi : array [0..(10*undo_maks)-1] of  
             array [0..3] of string  
current_login : array [0..2] of string
```

**ALGORITMA**

```
{ import modul dan fungsi buatan }  
import os  
from tools import cari_index_username, data_remove, data_append,  
                  int_min, int_maks, int_join,  
                  generate_bahan_bangunan, string_split,  
                  string_strip, string_append,  
                  string_in_array, string_leksikografis_min,  
                  string_leksikografis_maks, cari_index_candi,  
                  matrix_str_join, candi_append  
from data import data_save, data_load, Data  
  
{ Inisialisasi variabel }  
current_login <- ["", "", ""]  
undo_maks <- 100  
{undo_jin dan undo_candi diisi string kosong ""}
```

Di bawah ini adalah fungsi dan prosedur yang terdapat di dalamnya:

**procedure run**

```
(input command: string,  
 input/output users: Data,  
 input/output candi: Data,  
 input/output bahan_bangunan : Data)  
{Membaca masukkan dari user dan menjalankan command tersebut  
berdasarkan akun login saat ini}
```

**KAMUS LOKAL**

```
length : integer  
command_list : array [0..length-1] of string
```



**ALGORITMA**

```
command <- string_strip(command)
{ Agar command bersih dari spasi kosong }
depend on (command, current_login)
  command = "login" :
    current_login <- login(current_login, users)
  command = "logout" :
    current_login <- logout(current_login)
  command = "help" :
    help(current_login)
  command = "exit" :
    exit(users, candi, bahan_bangunan)
  command = "save" :
    save(users, candi, bahan_bangunan)
  current_login = ["", "", ""] :
    command_list <- ["logout", "summonjin", "hapusjin",
                     "ubahjin", "bangun", "kumpul",
                     "batchkumpul", "batchbangun",
                     "laporanjin", "laporancandi",
                     "hancurkancandi", "ayamberkokok",
                     "undohapus" ]

    length <- 13
    if( any (command = command_list[i] for i in
              range(length))) then
      output(command, " gagal!")
      output("Anda perlu login terlebih dahulu!")
    else
      output("Command \"", command, "\" tidak
              dikenal.")
  else
    depend on (command)
    command = "summonjin" :
      if(current_login[2] = "bandung_bondowoso")
      then
        summonjin(users)
      else
        output("Summonjin hanya dapat diakses
                oleh akun Bandung Bondowoso.")
    command = "hapusjin":
      if(current_login[2] =
        "bandung_bondowoso") then
        hapusjin(users, candi, undo_jin,
        undo_candi)
      else
```

```

        output("Hapusjin hanya dapat diakses
        oleh Bandung Bondowoso.")
command = "undohapus":
    if(current_login[2] =
    "bandung_bondowoso") then
        undohapus(users,candi, undo_jin,
        undo_candi)
    else
        output("Undohapus hanya dapat diakses
        oleh Bandung Bondowoso.")
command = "ubahjin":
    if(current_login[2] =
    "bandung_bondowoso") then
        ubahjin(users)
    else
        output("Ubahjin hanya dapat diakses
        oleh
        Bandung Bondowoso.")
command = "bangun":
    if(current_login[2] = "jin_pembangun") then
        bangun(candi, bahan_bangunan,
        current_login[0]], 1, 1, False)
    else
        output("Bangun hanya dapat diakses oleh
        akun Jin Pembangun")
command = "kumpul":
    if(current_login[2] = "jin_pengumpul") then
        kumpul(bahan_bangunan,1,False)
    else
        output("Kumpul hanya dapat diakses oleh
        akun Jin Pengumpul")
command = "batchkumpul":
    if(current_login[2] =
    "bandung_bondowoso") then
        batchkumpul(users,bahan_bangunan)
    else
        output("Batchkumpul hanya dapat diakses
        oleh
        Bandung Bondowoso.")
command = "batchbangun":
    if(current_login[2] =
    "bandung_bondowoso") then
        batchbangun(users,candi,bahan_bangunan)
    else

```

```

        output("Batchbangun hanya dapat diakses
        oleh Bandung Bondowoso.")
command = "laporanjin":
    if(current_login[2] =
    "bandung_bondowoso") then
        laporanjin(users,candi,bahan_bangunan)
    else
        output("Laporan jin hanya dapat diakses
        oleh Bandung Bondowoso.")
command = "laporancandi":
    if(current_login[2] =
    "bandung_bondowoso") then
        laporancandi(candi)
    else
        output("Laporan candi hanya dapat
        diakses oleh Bandung Bondowoso.")
command = "hancurkancandi":
    if(current_login[2] = "roro_jonggrang") then
        hancurkancandi(candi)
    else
        output("Hancurkan candi hanya dapat
        diakses oleh akun Roro Jonggrang.")
command = "ayamberkokok":
    if(current_login[2] = "roro_jonggrang") then
        ayamberkokok(candi)
    else
        output("Ayamberkokok hanya dapat
        diakses oleh akun Roro Jonggrang.")
else
    output("command \"", command, "\"" tidak
    dikenal.")

```

**{ F01 - Login }**

**function login**

```

    (current_login: array [0..2] of string,
    users: Data ) -> array [0..2] of string
{ Melakukan aksi login dan mengembalikan data hasil login yaitu
array string [username,password,role]}

```

**KAMUS LOKAL**

```

index : integer
username, password : string
isi_users : array[0..users.n_baris-1] of

```

array[0..users.n\_kolom-1] of string

**ALGORITMA**

```
if (current_login ≠ ["", "", ""]) then
{ user sudah login }
  output ("Login gagal!")
  output ("Anda telah login dengan username",
    current_login[0], "silahkan lakukan \"logout\"
    sebelum melakukan login kembali.")
  -> current_login
else {current_login = ["", "", ""]}
{ user belum login }
isi_users <- users.isi
input(username, password)
index <- cari_index_username(users,username)
{ index = -1 artinya username tidak ditemukan }
if (index ≠ -1) then
  if (password = isi_users[index][1]) then
    output("Selamat datang, ", username, "!")
    output("Masukkan command \"help\" untuk daftar
      command yang dapat kamu panggil")
    -> isi_users[index]
  else {password ≠ isi_users[index][1]}
    output("Password salah!")
    -> ["", "", ""]
  else {index = -1}
    output("Username tidak terdaftar!")
    -> ["", "", ""]
```

**{ F02 - Logout }**

**function logout**

```
(current_login: array [0..2] of string)
-> array [0..2] of string
{Melakukan logout akun dengan cara mengosongkan data
current_login}
```

**KAMUS LOKAL**

current\_login : array [0..2] of string

**ALGORITMA**

```
if (current_login = ["", "", ""]) then
  output("Logout gagal!")
  output("Anda belum login, silahkan login terlebih
```

```

        dahulu
            sebelum melakukan logout")
    -> current_login
else {current_login = ["", "", ""]}
    output("Logout berhasil!")
    -> ["", "", ""]

```

### **{ F03 - Summon Jin }**

**procedure** summonjin(input/output users: Data)  
 {Mensummon atau membuat jin baru}

#### **KAMUS LOKAL**

```

n_baris, i : integer
jenis_jin, username, password, role : string
username_valid : boolean
isi_users : array [0..users.n_baris-1] of
                array [0..users.n_kolom-1] of string

```

#### **ALGORITMA**

```

isi_users <- users.isi
n_baris <- users.n_baris
if(users.n_baris = 102) then
    output("Jumlah Jin telah maksimal! (100 jin). Bandung tidak
        dapat men-summon lebih dari itu")
else {jumlah jin < 100}
    output("Jenis jin yang dapat dipanggil:")
    output(" (1) Pengumpul - Bertugas mengumpulkan bahan
        bangunan")
    output(" (2) Pembangun - Bertugas membangun candi")
    while True do
        input(jenis_jin)
        output("")
    depend on (jenis_jin)
    jenis_jin = "1" : role <- "jin_pengumpul"
                    output("Memilih jin \"Pengumpul\".")
                    break

    jenis_jin = "2" : role <- "jin_pembangun"
                    output("Memilih jin \"Pembangun\".")
                    break

    else
        output("Tidak ada jenis jin bernomor \""+ jenis_jin
            + "\"!")
    while True do

```

```

{ ASUMSI : username alphanumeric (hanya terdiri dari angka
dan huruf)}
  input(username)
  if(username[0] = " ") then
    output("Username tidak boleh dimulai dengan spasi
    (\ " \")")
    continue
  username_valid <- True
  {menentukan apakah username sudah diambil atau belum}
  i traversal[0..n_baris-1]
    if(isi_users[i][0] = username)then
      output("\\nUsername \", username, "\\\" sudah
      diambil!")
      username_valid <- False
      break
    if(username_valid)then
      break
  {membuat password yang valid}
  iterate
    input(password)
    if(5 ≤ len(password) ≤ 25)then
      output()
      break
    else
      output("\\nPassword panjangnya harus 5-25
      karakter!")
    output()
  until (len(password)<5 or len(password)>25)
  {menambahkan data jin}
  users <- data_append(users,[username,password,role])
  output("Mengumpulkan sesajen...\\nMenyerahkan
  sesajen...\\nMembacakan mantra...")
  output("\\nJin ", username, "berhasil dipanggil!")

```

#### **{ F04 - Hilangkan Jin }**

##### **procedure hapusjin**

```

(input/output users: Data,
input/output candi: Data,
input/output undo_jin: array[0..2*undo_maks] of
array[0..3] of string,
input/output undo_candi: array[0..10*undo_maks] of
array[0..3] of string)
{Menghapus jin serta candi yang dibuatnya}

```

**KAMUS LOKAL**

```
n_baris_candi, count_candi_hapus, i, j, index : integer
username, jawab : string
candi_dihapuskan : < isi : array[0..length-1] of integer,
                    length: integer >
isi_candi : array [0..candi.n_baris-1] of
              array [0..candi.n_kolom-1] of string
```

**ALGORITMA**

```
input(username)
index <- cari_index_username(users,username)
if(index = -1) then
    output("Tidak ada jin dengan username tersebut.")
else {index ≠ -1}
    input(jawab)
    if(jawab = "Y") then
        candi_dihapuskan <- cari_index_candi(candi,username)
        count_candi_hapus <- candi_dihapuskan[1]
        {Menyimpan candi yang dibuat oleh jin tersebut}
        i traversal [count_candi_hapus-1..0]
            j traversal [0..(10*undo_maks)-1]
                if(undo_candi[j][0] = "") then
                    undo_candi[j] <-
                        [candi.isi[candi_dihapuskan[0][i]][1],
                        candi.isi[candi_dihapuskan[0][i]][2],
                        candi.isi[candi_dihapuskan[0][i]][3],
                        candi.isi[candi_dihapuskan[0][i]][4] ]
                    break
        {Menghapus candi- candi}
        i traversal [0..count_candi_hapus-1]
            candi <- data_remove(candi,
                                candi_dihapuskan[0][i]-i)
        {Menyimpan data jin yang akan dihapus}
        i traversal [0..(2*undo_maks)-1]
            if (undo_jin[i][0] = "") then
                undo_jin[i] <-
                    string_append(users.isi[index],
                                str(count_candi_hapus),3)
                break
        {Menghapus data jin yang dihapus}
        users <- data_remove(users,index)
        output("\\nJin telah berhasil dihapus dari alam gaib.")
    else if(jawab ≠ "N") then
        output("Tidak ada opsi \\\"", jawab, "\\\". Ulangi!")
        hapusjin(users,candi)
```

<pre> { <b>B04 - Undo</b> } <b>procedure undohapus</b>   (input/output users: <u>Data</u>,    input/output candi: <u>Data</u>,    input/output undo_jin: array [0..(2*undo_maks)-1] of                                    array [0..3] of <u>string</u>,    input/output undo_candi: array [0..(10*undo_maks)-1] of                                    array [0..3] of <u>string</u> ) {Mengembalikan keadaan jin serta candi hasil bangunnya yang telah dihapus} </pre>
<p><b>KAMUS LOKAL</b></p> <pre> i, j, count_undo_candi : <u>integer</u> username : <u>string</u> </pre>
<p><b>ALGORITMA</b></p> <pre> <u>if</u>(undo_jin[0] = ["","","",""]) <u>then</u>   <u>output</u>("Undo gagal!")   <u>output</u>("Tidak ada jin yang sudah dihapus") <u>else</u>   i <u>traversal</u> [0..(2*undo_maks)-1,-1,-1]     <u>if</u>(undo_jin[i][0] ≠ "") <u>then</u>       count_undo_candi &lt;- int(undo_jin[i][3])       username &lt;- undo_jin[i][0]       users &lt;- data_append(users, [undo_jin[i][0],                                    undo_jin[i][1],                                    undo_jin[i][2]] )       undo_jin[i] &lt;- ["","","",""]       <u>break</u>   i <u>traversal</u> [0..count_undo_candi]     j <u>traversal</u> [(10 * undo_maks)-1,-1,-1)..0]       <u>if</u>(undo_candi[j][0] ≠ "") <u>then</u>         candi_append(candi, undo_candi[j])         undo_candi[j] &lt;- ["","","",""]         <u>break</u>   <u>output</u>("Undo berhasil!")   <u>output</u>("Jin\""", username, "\"sudah kembali dengan",         count_undo_candi, "candi") </pre>
<pre> { <b>F05 - Ubah Tipe Jin</b> } <b>procedure ubahjin</b> ( input/output users: <u>Data</u> ) </pre>



```
{Mengubah role dari jin, jika role pembangun maka dapat diubah ke pengumpul dan sebaliknya}
```

#### **KAMUS LOKAL**

```
n_baris_candi : integer  
bahan_dimiliki, bahan_diperlukan : array[0..2] of integer  
data_hasil : array[0..3] of string  
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of  
array [0..bahan_bangunan.n_kolom-1] of  
string
```

#### **ALGORITMA**

```
isi_users <- users.isi  
input(username)  
index <- cari_index_username(users,username)  
if(index ≠ -1) then  
    if(isi_users[index][2] = "jin_pengumpul") then  
        input(ubah)  
        if(ubah = "Y") then  
            isi_users[index][2] <- "jin_pembangun"  
            output("Jin telah berhasil diubah.")  
        else {ubah ≠ "Y"}  
            output("Jin tidak diubah.")  
    else { index = -1}  
        input(ubah)  
        if(ubah = "Y") then  
            isi_users[index][2] <- "jin_pengumpul"  
            output("Jin telah berhasil diubah.")  
        else { ubah = "Y" }  
            output("jin tidak diubah.")  
    else {username tidak ditemukan}  
        output("\\nTidak ada jin dengan username tersebut.")  
users.isi <- isi_users
```

#### **{ F06 - Jin Pembangun }**

##### **function bangun**

```
(candi : Data,  
bahan_bangunan : Data,  
jin_pembangun : array [0..length-1] of string,  
length : integer,  
jumlah_loop : integer,  
bangun_batch : boolean )  
-> array [0..length-1] of array [0..3] of string  
{Membangun candi dengan bahan bangunan yang acak, jika jumlah
```

```
jin_pembangun > 1 maka pembangunan dilakukan secara rekursif}
```

#### KAMUS LOKAL

```
n_baris_candi : integer  
bahan_dimiliki, bahan_diperlukan : array[0..2] of integer  
data_hasil : array[0..3] of string  
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of  
array [0..bahan_bangunan.n_kolom-1] of  
string
```

#### ALGORITMA

```
isi_candi <- candi.isi  
isi_bahan_bangunan <- bahan_bangunan.isi  
n_baris_candi <- candi.n_baris  
{bahan dalam format [pasir,batu,air]}  
bahan_dimiliki <- [int(isi_bahan_bangunan[0][2]),  
int(isi_bahan_bangunan[1][2]),  
int(isi_bahan_bangunan[2][2]) ]  
bahan_diperlukan <- generate_bahan_bangunan()  
data_hasil <- [jin_pembangun[length-jumlah_loop],  
str(bahan_diperlukan[0]),  
str(bahan_diperlukan[1]),  
str(bahan_diperlukan[2]) ]  
if(bangun_batch = False) then  
  if (bahan_dimiliki[0] < bahan_diperlukan[0] or  
    bahan_dimiliki[1] < bahan_diperlukan[1] or  
    bahan_dimiliki[2] < bahan_diperlukan[2]) then  
    { Candi tidak dibangun }  
    output("# Men-generate bahan bangunan",  
      (bahan_diperlukan[0], "pasir",  
        bahan_diperlukan[1], "batu, dan",  
        "bahan_diperlukan[2] air)" )  
    output("# Memiliki ", bahan_dimiliki[0], "pasir, ",  
      bahan_dimiliki[1], "batu, ",  
      bahan_dimiliki[2], "air" )  
    output("Bahan bangunan tidak mencukupi.")  
    output("Candi tidak bisa dibangun!")  
  else {Candi dibangun}  
  {Mengurangi bahan bangunan yang dimiliki}  
  isi_bahan_bangunan[0][2] <-  
    str(int(isi_bahan_bangunan[0][2]) -  
      bahan_diperlukan[0])  
  isi_bahan_bangunan[1][2] <-  
    str(int(isi_bahan_bangunan[1][2]) -  
      bahan_diperlukan[1])
```

```

isi_bahan_bangunan[2][2] <-
  str(int(isi_bahan_bangunan[2][2]) -
    bahan_diperlukan[2])
bahan_bangunan.isi <- isi_bahan_bangunan
if(n_baris_candi < 100) then
  candi_append(candi,data_hasil)
  output("Candi berhasil dibangun.")
  output("Sisa candi yang perlu dibangun:",
    (100-n_baris_candi) )
else
  output("Candi berhasil dibangun.")
  output("Sisa candi yang perlu dibangun: 0")
if(jumlah_loop ≠ 1) then
  -> matrix_str_join(data_hasil, bangun(candi,
    bahan_bangunan, jin_pembangun,
    length, jumlah_loop-1, True),
    jumlah_loop-1, 4)
else
  -> [data_hasil]

```

#### **{ F07 - Jin Pengumpul }**

##### **function kumpul**

```

(bahan_bangunan : Data,
 jumlah_loop : integer,
 kumpul_batch : boolean )-> array [0..2] of string
{Mengumpulkan bahan bangunan yang jumlahnya secara acak, untuk batch
kumpul dilakukan dengan rekursif}

```

##### **KAMUS LOKAL**

```

bahan_dikumpul : array[0..2] of integer
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of
  array [0..bahan_bangunan.n_kolom-1] of
    string

```

##### **ALGORITMA**

```

isi_bahan_bangunan <- bahan_bangunan.isi
bahan_dikumpul <- generate_bahan_bangunan()
{Menambahkan bahan bangunan yang dikumpul ke bahan bangunan
total}
isi_bahan_bangunan[0][2] <- str(int(isi_bahan_bangunan[0][2]) +
  bahan_dikumpul[0])
isi_bahan_bangunan[1][2] <- str(int(isi_bahan_bangunan[1][2]) +
  bahan_dikumpul[1])

```

```

isi_bahan_bangunan[2][2] <- str(int(isi_bahan_bangunan[2][2]) +
                                bahan_dikumpul[2])
bahan_bangunan.isi <- isi_bahan_bangunan
depend on (kumpul_batch, jumlah_loop)
    kumpul_batch = False : output("Jin menemukan",
bahan_dikumpul[0], "pasir", bahan_dikumpul[1], "batu, dan",
bahan_dikumpul[2], "air.")

    jumlah_loop ≠ 1 : -> int_join(bahan_dikumpul,
    kumpul(bahan_bangunan, jumlah_loop-1, True), 3)
else
    -> bahan_dikumpul

```

#### **{ F08 - Batch Kumpul }**

##### **procedure batchkumpul**

```

(input users: Data,
input/output bahan_bangunan: Data )
{Melakukan pengumpulan bahan bangunan oleh semua jin pengumpul yang
ada}

```

##### **KAMUS LOKAL**

```

i, n_baris_users, count_jin_pengumpul : integer
hasil_kumpul : array[0..2] of integer
isi_users : array [0..users.n_baris-1] of
array [0..users.n_kolom-1] of string

```

##### **ALGORITMA**

```

isi_users <- users.isi
n_baris_users <- users.n_baris
{Hitung jumlah jin pengumpul}
count_jin_pengumpul <- 0
i traversal [0..n_baris_users]
    if(isi_users[i][2] = "jin_pengumpul")
        count_jin_pengumpul <- count_jin_pengumpul + 1
if(count_jin_pengumpul = 0) then
    output("Kumpul gagal. Anda tidak punya jin pengumpul.
    Silahkan summon terlebih dahulu."
else
    output("Mengerahkan", count_jin_pengumpul, "jin untuk
    mengumpulkan bahan."
    {Mengumpulkan bahan}
    hasil_kumpul <- kumpul(bahan_bangunan,
        count_jin_pengumpul, True)

```

```

        output("Jin menemukan total", hasil_kumpul[0], "pasir",
            hasil_kumpul[1], "batu, dan", hasil_kumpul[2],
            "air.")
    
```

#### **{ F08 - Batch Bangun }**

##### **procedure batchbangun**

```

    (input users: Data,
     input/output candi: Data,
     input/output bahan_bangunan: Data )
    {Melakukan pembangunan candi oleh setiap jin pembangun yang ada}
    
```

##### **KAMUS LOKAL**

```

    i, index, n_baris_users, count_jin_pembangun, total_pasir,
    total_batu, total_air, kurang_pasir, kurang_batu,
    kurang_air : integer

    bahan_dimiliki : array[0..2] of integer
    jin_pembangun : array[0..count_jin_pembangun-1] of string
    isi_users : array [0..users.n_baris-1] of
        array [0..users.n_kolom-1] of string
    isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of
        array [0..bahan_bangunan.n_kolom-1] of
            string
    data_pembangunan : array [0..count_jin_pembangun-1] of
        array [0..3] of string
    
```

##### **ALGORITMA**

```

    isi_users <- users.isi
    isi_bahan_bangunan <- bahan_bangunan_isi
    n_baris_users <- users.n_baris
    bahan_dimiliki <- [int(isi_bahan_bangunan[0][2],
                        int(isi_bahan_bangunan[1][2],
                        int(isi_bahan_bangunan[2][2]))]
    {Hitung jumlah jin pembangun}
    count_jin_pembangun <- 0
    i traversal [0..n_baris_users]
        if(isi_users[i][2] = "jin_pembangun") then
            count_jin_pembangun <- count_jin_pembangun + 1
    if(count_jin_pembangun = 0) then
        output("Bangun gagal. Anda tidak punya jin pembangun.
            Silahkan summon terlebih dahulu.")
    else {ada jin_pembangun}
        {Mengumpulkan nama semua jin pembangun}
    
```

```

jin_pembangun <- [""] for_in range(count_jin_pembangun)]
index = 0
i traversal [0..n_baris_users]
  if(isi_users[i][2] = "jin_pembangun"):
    jin_pembangun[index] <- isi_users[i][0]
    index <- index + 1
{Menghitung jumlah candi yang sudah dibangun}
data_pembangunan <- bangun(candi, bahan_bangunan,
                           jin_pembangun, count_jin_pembangun,
                           count_jin_pembangun, True)

total_pasir <- 0
total_batu <- 0
total_air <- 0
i traversal [0..count_jin_pembangun]
  total_pasir <- total_pasir +
    int(data_pembangunan[i][1])
  total_batu <- total_batu +
    int(data_pembangunan[i][2])
  total_air <- total_air +
    int(data_pembangunan[i][3])
output("Mengerahkan",count_jin_pembangun,"jin untuk
membangun candi dengan total bahan",
total_pasir, "pasir", total_batu, "batu, dan",
total_air, "air." )
if(total_pasir > bahan_dimiliki[0] or
total_batu > bahan_dimiliki[1] or
total_air > bahan_dimiliki[2]) then
  kurang_pasir <- 0
  kurang_batu <- 0
  kurang_air <- 0
  depend_on (total_pasir, total_batu,
bahan_dimiliki[i])
    total_pasir > bahan_dimiliki[0]:
      kurang_pasir <- total_pasir -
        bahan_dimiliki[0]
      output(kurang_pasir, "pasir")
    total_batu > bahan_dimiliki[1]:
      kurang_batu <- total_batu -
        bahan_dimiliki[1]
    total_batu > bahan_dimiliki[2]:
      kurang_air <- total_air - bahan_dimiliki[2]
  kurang_pasir ≠ 0 and kurang_batu ≠ 0 and
  kurang_air ≠ 0 :
    output("Bangun gagal. Kurang",

```

```

        kurang_pasir, "pasir", kurang_batu, "batu,
        dan", kurang_air, "air.")
    kurang_pasir ≠ 0 and kurang_batu ≠ 0:
        output("Bangun gagal. Kurang",
        kurang_pasir, "pasir dan", kurang_batu,
        "batu.")
    kurang_pasir ≠ 0 and kurang_air ≠ 0:
        output("Bangun gagal. Kurang",
        kurang_pasir, "pasir dan", kurang_air,
        "air.")
    kurang_batu ≠ 0 and kurang_air ≠ 0 :
        output("Bangun gagal. Kurang", kurang_batu,
        "batu, dan", kurang_air,"air.")
    kurang_pasir ≠ 0 :
        output("Bangun gagal. Kurang",
        kurang_pasir,
        " pasir.")
    kurang_batu ≠ 0
        output("Bangun gagal. Kurang", kurang_batu,
        "batu.")
    else {kurang_air ≠ 0}
        output("Bangun gagal. Kurang", kurang_air,
        "air.")
    else {bahan bangunan memenuhi bahan yang diperlukan
    untuk membangun}
    isi_bahan_bangunan[0][2] <-
    str(int(isi_bahan_bangunan[0][2]) -
    total_pasir)
    isi_bahan_bangunan[1][2] <-
    str(int(isi_bahan_bangunan[1][2]) - total_batu)
    isi_bahan_bangunan[2][2] <-
    str(int(isi_bahan_bangunan[2][2]) - total_air)
    bahan_bangunan.isi <- isi_bahan_bangunan
    i traversal [0..count_jin_pembangun]
        if(candi.n_baris < 100) then
            candi_append(candi,data_pembangunan[i])
    output("jin berhasil membangun total",
    count_jin_pembangun, "candi.")

```

**{ F09 - Ambil Laporan Jin }**

**procedure laporanjin**

```

    (input users: Data,
    input candi: Data,
```

```

    input bahan_bangunan: Data)
{Melakukan laporan jin mulai dari total jin masing-masing jenis, jin
termalas dan terajin serta bahan bangunan}

```

#### **KAMUS LOKAL**

```

n_baris_users, n_baris_candi, total_jin, total_pengumpul,
total_pembangun, i, index, n_jin, count_maks, count_min,
candi_maks candi_min, index_jin_maks,
index_jin_min : integer

count_candi_jin : array [0..n_jin-1] of integer
jin_maks : array [0..count_maks-1] of string
jin_min : array [0..count_min-1] of string
jin : array [0..total_pembangun-1] of string
isi_users : array [0..users.n_baris-1] of
            array [0..users.n_kolom-1] of string
isi_bahan_bangunan : array [0..bahan_bangunan.n_baris-1] of
            array [0..bahan_bangunan.n_kolom-1] of
            string
data_pembangunan : array [0..count_jin_pembangun-1] of
            array [0..3] of string

```

#### **ALGORITMA**

```

isi_users <- users.isi
isi_candi <- candi.isi
isi_bahan_bangunan <- bahan_bangunan.isi
n_baris_users <- users.n_baris
n_baris_candi <- candi.n_baris
{Menentukan jumlah jin dan tipenya}
total_jin <- 0
total_pengumpul <- 0
total_pembangun <- 0
i traversal [0..n_baris_users]
    if(isi_users[i][2] = "jin_pengumpul") then
        total_jin <- total_jin + 1
        total_pengumpul <- total_pengumpul + 1
    if(isi_users[i][2] = "jin_pembangun") then
        total_jin <- total_jin + 1
        total_pembangun <- total_pembangun + 1
output("Total Jin:", total_jin)
output("Total Jin Pengumpul:", total_pengumpul)
output("Total Jin Pembangun:", total_pembangun)

{Mencari jin termalas dan terajin}

```



```

if (n_baris_candi = 0) then
  output("> Jin Terajin : -")
  output("> Jin Termalas : -")
else
  {mencari semua jin pembangun}
  jin <- [" " for_in range(total_pembangun)]
  n_jin <- total_pembangun
  index <- 0
  i traversal [0..n_baris_users]
    if(isi_users[i][2] = "jin_pembangun") then
      jin[index] <- isi_users[i][0]
      index <- index + 1
  {memastikan semua pembuat candi ada dalam array jin,
  jika tidak maka ditambahkan}
  i traversal [0..n_baris_candi]
    index <- string_in_array(jin, isi_candi[i][1],
                             n_jin )
    if(index = -1) then
      jin <- string_append(jin, isi_candi[i][1],
                           n_jin)
      n_jin = n_jin + 1
  count_candi_jin <- [0 for_in range(n_jin)]
  i traversal [0..n_baris_candi]

```

```

        if(index ≠ -1) then
            count_candi_jin[index] <- count_candi_jin[index]
                                + 1
{ mencari maksimum dan minimum jumlah candi yang dibuat
oleh seorang jin }
candi_maks <- int_maks(count_candi_jin,n_jin)
candi_min <- int_min(count_candi_jin, n_jin)
{ Menghitung jumlah jin yang membangun candi dengan jumlah
maksimum atau minimum }
count_maks <- 0
count_min <- 0
i traversal[0..n_jin]
    if(count_candi_jin[i] = candi_maks)then
        count_maks <- count_maks + 1
    if(count_candi_jin[i] = candi_min) then
        count_min <- count_min + 1
{Mencari semua jin yang membangun candi dengan jumlah
maksimum atau minimum ke dalam array of string}
jin_maks <- ["" for_in range(count_maks)]
jin_min <- ["" for_in range(count_min)]
index_jin_maks <- 0
index_jin_min <- 0
i traversal [0..n_jin]
    if(count_candi_jin[i] = candi_maks) then
        jin_maks[index_jin_maks] <- jin[i]
        index_jin_maks <- index_jin_maks + 1
    if(count_candi_jin[i] = candi_min) then
        jin_min[index_jin_min] <- jin[i]
        index_jin_min <- index_jin_min + 1
{Mengembalikan jin termalas dan terajin berdasarkan
leksikografis}
output("> Jin Terajin: ",
        string_leksikografis_min(jin_maks,count_maks))
output("> Jin Termalas: ",
        string_leksikografis_min(jin_min,count_min))
{Menentukan jumlah bahan bangunan}
output("> Jumlah Pasir : ",
isi_bahan_bangunan[0][2],"unit")
output("> Jumlah Batu : ", isi_bahan_bangunan[1][2],"unit")
output("> Jumlah Air : ", isi_bahan_bangunan[2][2],"unit")

```

#### **{ F10 - Ambil Laporan Candi }**

**procedure** laporcandicandi( input candi: Data)

{Program ini berfungsi untuk melakukan laporan candi mulai dari jumlah candi, bahan total yang diperlukan, serta candi termurah dan termahal}

**KAMUS LOKAL**

i, n\_baris\_candi, harga\_maks, harga\_min, total\_pasir,  
total\_batu, total\_air : integer

harga : array [0..n\_baris\_candi-1] of integer

isi\_candi : array [0..candi.n\_baris-1] of  
array [0..candi.n\_kolom-1] of string

**ALGORITMA**

```
isi_candi <- candi.isi
n_baris_candi <- candi.n_baris
{Menghitung bahan bangunan total pembuat semua candi}
total_pasir <- 0
total_batu <- 0
total_air <- 0
i traversal [0..n_baris_candi]
    total_pasir <- total_pasir + int(isi_candi[i][2])
    total_batu <- total_batu + int(isi_candi[i][3])
    total_air <- total_air + int(isi_candi[i][4])
output("> Total Candi: ", n_baris_candi)
output("> Total Pasir yang digunakan: ", total_pasir)
output("> Total Batu yang digunakan: ", total_batu)
output("> Total air yang digunakan: ", total_air)
{Menghitung harga tiap candi}
harga <- [0 for_in range(n_baris_candi)]
i traversal[0..n_baris_candi]
    if(harga[i] > harga_maks) then
        harga_maks <- harga[i]
    if(harga[i] < harga_min) then
        harga_min <- harga[i]
{Mengeluarkan candu dengan harga termahal atau termurah
dengan index paling kecil}
i traversal[0.._baris_candi]
    if(harga[i] = harga_maks) then
        output("> ID Candi Termahal: ", isi_candi[i][0],
            "Rp", harga_maks)
        break
i traversal[0..n_baris_candi]
    if(harga[i] = harga_min) then
        output("> ID Candi Termurah: ", isi_candi[i][0],
            "Rp", harga_min)
        break
```

<pre> { <b>F11 - Hancurkan Candi</b> } <b>procedure</b> hancurkancandi( input/output candi: <u>Data</u>) {program ini berfungsi untuk menghancurkan candi dengan id yang diberikan } </pre>
<p><b>KAMUS LOKAL</b></p> <pre> i, index, n_baris_candi : <u>integer</u> id, yakin : <u>string</u> isi_candi : array [0..candi.n_baris-1] of array [0..candi.n_kolom-1] of <u>string</u> </pre>
<p><b>ALGORITMA</b></p> <pre> isi_candi &lt;- candi.isi n_baris_candi &lt;- candi.n_baris <u>input</u>(id) index &lt;- -1 i <u>traversal</u> (n_baris_candi)   <u>if</u>(isi_candi[i][0] = id) <u>then</u>     index &lt;- i     <u>break</u> <u>if</u> (index ≠ -1) <u>then</u>   <u>input</u>(yakin)   <u>if</u> (yakin = "Y") <u>then</u>     {candi dihancurkan}     candi &lt;- data_remove(candi,index)     <u>output</u>("Candi telah berhasil dihancurkan.")   <u>else if</u> (yakin ≠ "N") <u>then</u>     <u>output</u>("Tidak ada opsi", "\"yakin\".", "Ulangi!")     hancurkancandi(candi) <u>else</u>   <u>output</u>("Tidak ada candi dengan ID tersebut.") </pre>
<pre> { <b>F12 - Ayam Berkokok</b> } <b>procedure</b> ayamberkokok (input candi: <u>Data</u>) {program ini berfungsi untuk menentukan pemenang dari game dengan cara memasukkan pagi hari} </pre>
<p><b>KAMUS LOKAL</b></p> <pre> i, banyak_candi : <u>integer</u> </pre>
<p><b>ALGORITMA</b></p> <pre> banyak_candi &lt;- candi.n_baris <u>output</u>("Kukuruyuk.. Kukuruyuk..") <u>output</u>("Jumlah Candi:", banyak_candi) <u>if</u>(banyak_candi ≥ 100) <u>then</u> </pre>

```

        output("Yah, Bandung Bondowoso memenangkan permainan!")
    else
        output("Selamat, Roro Jonggrang memenangkan permainan!")
        output("Bandung Bondowoso angry noise")
        output("Roro Jonggrang dikutuk menjadi candi.")
quit()

```

### **{ F13 - Load (Dengan argparse) }**

**function load()** -> array [0..2] of Data

{Program ini menggunakan argparse agar dapat meload / membuka kembali data yang sudah disave sebelumnya, fungsi ini sendiri dijalankan hanya sekali saja pada command line / terminal}

### **KAMUS LOKAL**

```

parser, args : objek dari modul argparse
args.nama_folder : string
users, candi, bahan_bangunan : Data

```

### **ALGORITMA**

```

parser <- argparse.ArgumentParser(add_help=False, usage='% (prog) s
                                <nama_folder>')
parser.add_argument("nama_folder", nargs = "?", type = str,
                    default = "")
args <- parser.parse_args()
path <- "save/"+args.nama_folder
if(path = "save/") then
    output("Tidak ada nama folder yang diberikan!\n")
    parser.print_usage()
    -> []
else if(os.path.exists(path)) then
    output("Loading...")
    users <- data_load(path+"/user.csv")
    candi <- data_load(path+"/candi.csv")
    bahan_bangunan <- data_load(path+"/bahan_bangunan.csv")
    output("Selamat datang di program \"Manajerial Candi\"!")
    output("Masukkan command \"help\" untuk daftar command yang
        dapat kamu panggil.\n")
    -> [users,candi,bahan_bangunan]
else {Folder tidak ditemukan}
    output("Folder","\path\","tidak ditemukan")
    -> []

```

### **{ F14 - Save }**

**procedure save**

<pre> (input users: <u>Data</u>,  input candi: <u>Data</u>,  input bahan_bangunan: <u>Data</u>) {Menyimpan data pada folder dengan parent folder "save"} </pre>
<p><b>KAMUS LOKAL</b></p> <pre> count_splitted, i : <u>integer</u> path, current_path : <u>string</u> file_baru : <u>boolean</u> path_splited : array [0..count_splitted-1] of <u>string</u> </pre>
<p><b>ALGORITMA</b></p> <pre> path &lt;- "save/" + <u>input</u>(nama_folder) <u>output</u>("Saving...") path_splited &lt;- string_split(path, "/") current_path &lt;- "" file_baru &lt;- False dir <u>traversal</u> [0..path_splited]   current_path &lt;- current_path + dir   <u>if</u>(not os.path.exists(current_path)) <u>then</u>     file_baru &lt;- True     <u>output</u>("Membuat folder",current_path,"...")     os.mkdir(current_path)   current_path &lt;- current_path + "/" <u>if</u>(file_baru) <u>then</u>   <u>output</u>() data_save(path, "user", users) data_save(path, "candi", candi) data_save(path, "bahan_bangunan", bahan_bangunan) <u>output</u>("Berhasil menyimpan data di folder",path+"!") </pre>
<p><b>{ F15 - Help }</b></p> <p><b><u>procedure help</u></b>( input current_login: array [0..2] of <u>string</u>)  {Memberikan list semua command yang dapat digunakan tergantung pada status login}</p>
<p><b>KAMUS LOKAL</b></p> <pre> current_login : array[0..2] of <u>string</u> </pre>
<p><b>ALGORITMA</b></p> <pre> <u>output</u>("HELP".center(28,"=)) <u>if</u>(current_login = ["for_in range(3)]) <u>then</u> { Belum Login }   <u>output</u>("1. login") </pre>

```

        output("    Untuk masuk menggunakan akun")
        output("2. save")
        output("    Untuk menyimpan data permainan saat ini")
        output("3. exit")
        output("    Untuk keluar dari program dari kembali ke
            terminal")
    else if(current_login[2] = "bandung_bondowoso") then {
Bandung Bondowoso }
        output("1. logout")
        output("    Untuk keluar dari akun yang digunakan
            sekarang")
        output("2. summonjin")
        output("    Untuk memanggil jin")
        output("3. hapusjin")
        output("    Untuk menghapus jin")
        output("4. undohapus")
        output("    Untuk mengembalikan jin yang terhapus")
        output("5. ubahjin")
        output("    Untuk mengubah tipe jin (pengumpul/pembangun)")
        output("6. batchkumpul")
        output("    Untuk melakukan pengumpulan oleh semua jin
            pengumpul")
        output("7. batchbangun")
        output("    Untuk melakukan pembangunan oleh semua jin
            pembangun")
        output("8. laporanjin")
        output("    Untuk mengambil laporan jin")
        output("9. laporan candi")
        output("    Untuk mengambil laporan candi")
        output("10. save")
        output("    Untuk menyimpan data permainan saat ini")
        output("11. exit")
        output("    Untuk keluar dari program dan kembali ke
            terminal")
    else if(current_login[2] = "roro_jonggrang") then { Roro
Jonggrang }
        output("1. logout")
        output("    Untuk keluar dari akun yang digunakan
            sekarang")
        output("2. hancurkancandi")
        output("    Untuk menghancurkan candi yang tersedia")
        output("3. ayamberkokok")
        output("    Untuk menyelesaikan permainan dengan memalsukan
            pagi hari")

```

```

        output("4. save")
        output("    Untuk menyimpan data permainan saat ini")
        output("5. exit")
        output("    Untuk keluar dari program dan kembali ke
            terminal")
    else if(current_login[2] = "jin_pengumpul") then { Jin
Pengumpul }
        output("1. logout")
        output("    Untuk keluar dari akun yang digunakan
            sekarang")
        output("2. kumpul")
        output("    Untuk mengumpulkan bahan bangunan")
        output("3. save")
        output("    Untuk menyimpan data permainan saat ini")
        output("4. exit")
        output("    Untuk keluar dari program dan kembali ke
            terminal")
    else { current_login[2] = "jin_pembangun" }
        output("1. logout")
        output("    Untuk keluar dari akun yang digunakan
            sekarang")
        output("2. bangun")
        output("    Untuk membangun sebuah candi")
        output("3. save")
        output("    Untuk menyimpan data permainan saat ini")
        output("4. exit")
        output("    Untuk keluar dari program dan kembali ke
            terminal")

```

#### **{ F16 - Exit }**

##### **procedure exit**

```

(input users: Data,
 input candi: Data,
 input bahan_bangunan: Data )

```

{ Program ini untuk keluar dari program, serta opsi menyimpan data sebelum keluar }

#### **KAMUS LOKAL**

jawab\_exit : string

#### **ALGORITMA**



```

    output("Apakah Anda mau melakukan penyimpanan file yang
           sudah diubah? (y/n)",end="")
    input(jawab_exit)
    depend on (jawab_exit)
        jawab_exit = "y" or jawab_exit = "Y"
        save(users, candi, bahan_bangunan)
        quit()
        jawab_exit = "n" or jawab_exit = "N"
        quit()
    else {Input salah sehingga prosedur diulang}
        exit(users, candi, bahan_bangunan)

```

## **Modul Numgen**

### **Module numgen**

{ Berisi fungsi untuk mendapatkan angka acak (random number)}

### **KAMUS**

seed : integer

### **ALGORITMA**

```

{ import modul time untuk mendapatkan waktu saat ini}
from time import time
seed <- int(time())

```

Di bawah ini adalah fungsi dan prosedur yang terdapat di dalamnya:

### **{ B01 - Random Number Generator }**

#### **function randomize**

```

    (lower_bound : integer,
     upper_bound : integer) -> integer
{ Mengembalikan angka random dengan algoritma Linear
  Congruential Generator (LCG)

```

### **KAMUS LOKAL**

```

const a : integer = 1583458089
const b : integer = 1132489760
m : integer

```

### **ALGORITMA**

```

a <- 1583458089

```

```

b <- 1132489760
m <- (2^31) - 1
seed <- (a*seed + b) mod m
while (seed mod (upper_bound + 1) < lower_bound) do
    seed <- (a*seed + b) mod m
-> seed mod (upper_bound + 1)

```

## Modul tools

### **Module tools**

{ Berisi semua fungsi buatan sebagai pelengkap}

### **KAMUS**

```

type Data : < isi : matriks of string,
              n_baris : integer,
              n_kolom : integer >

```

**Di bawah ini adalah fungsi dan prosedur yang terdapat di dalamnya:**

### **function string\_split**

```

(stringx : string, key : string) -> array [0..count-1] of string
{Fungsi ini untuk memisahkan suatu string "stringx" dengan pemisah
"key" dan mengembalikan array yang berisi hasil pemisahan}

```

### **KAMUS LOKAL**

```

count, i, index_str, index_hasil : integer
part : string
hasil : array [0..count-1] of string

```

### **ALGORITMA**

```

count <- 1
i traversal [0..len(stringx)-1]
    if(stringx[i] = key) then
        count <- count + 1
if(count = 1) then
    -> [stringx]
index_str <- 0

```

```

index_hasil <- 0
repeat
  if (index_str = len(stringx)) then
    break
  i traversal [index_str..len(stringx)-1]
    if(stringx[i] = key) then
      index_str <- index_str + 1
      break
    else
      index_str <- index_str + 1
      part <- part + stringx[i]
  hasil[index_hasil] <- part
  index_hasil <- index_hasil + 1
until index_str = len(stringx)
-> hasil

```

#### **function string\_slice**

```

(stringx : string, index_awal : integer, index_akhir : integer,
index_sekarang : integer = 0)-> string
{Fungsi ini untuk mengembalikan bagian stringx dari index_awal hingga
index_akhir}

```

#### **KAMUS LOKAL**

```

index_sekarang : integer

```

#### **ALGORITMA**

```

depend on(index_awal, index_sekarang, index_akhir, len(stringx))
  index_sekarang < index_awal : output(string_slice(stringx,
index_awal, index_akhir, index_sekarang+1)
  index_sekarang = len(stringx) : output("")
  index_sekarang = index_akhir-1 :
    output(stringx[index_sekarang])
else
  output(stringx[index_sekarang] + string_slice(stringx,
index_awal, index_akhir, index_sekarang+1)

```

#### **function string\_strip**

```

(stringx : string)-> array [index_awal..index_akhir-1] of string
{Fungsi ini untuk membersihkan string dari karakter kosong / spasi
(" ") pada awal dan akhir string}

```

#### **KAMUS LOKAL**

```

index_awal, index_akhir, i : integer

```

```
hasil : string
```

#### **ALGORITMA**

```
index_awal <- 0
i traversal [0..len(string)]
  if(string[i] ≠ "") then
    break
  else
    index_awal <- index_awal + 1
i traversal [range(len(string)-1, index_awal, -1..0)]
  if(string[i] ≠ "") then
    break
  else
    index_akhir <- index_akhir - 1

i traversal [0..range(index_awal, index_akhir)]
  hasil <- hasil + string[i]
-> hasil
```

#### **function string\_append**

```
(array : array [0..length-1] of string,
 stringx : string, length : integer)
-> array [0..length] of string
{Fungsi ini untuk menambahkan string ke dalam array of string}
```

#### **KAMUS LOKAL**

```
i : integer
array_baru : array [0..length] of string
```

#### **ALGORITMA**

```
i traversal [0..length-1]
  array_baru[i] <- array[i]
array_baru[length] <- stringx
-> array_baru
```

#### **function string\_in\_array**

```
(array : array [0..array_length-1] of string,
 stringx : string, array_length : integer) -> integer
{Fungsi ini untuk mengecek apakah string terdapat pada array jika
iya maka mengembalikan indexnya jika tidak ditemukan mengembalikan
nilai -1}
```

#### **KAMUS LOKAL**

i : <u>integer</u>
<b>ALGORITMA</b> i <u>traversal</u> (0..array_length-1] <u>if</u> (array[i] = stringx) <u>then</u> -> i -> -1
<b><u>function</u> string_leksikografis_min</b> (array : array [0..length-1] of <u>string</u> , length : <u>integer</u> ) -> <u>string</u> {Fungsi ini untuk menentukan string terkecil berdasarkan leksikografis}
<b>KAMUS LOKAL</b> i : <u>integer</u> min : <u>string</u>
<b>ALGORITMA</b> i <u>traversal</u> [0..length-1] <u>if</u> (array[i] < min) <u>then</u> min <- array[i] -> min
<b><u>function</u> string_leksikografis_maks</b> (array : array [0..length-1] of <u>string</u> , length : <u>integer</u> ) -> <u>string</u> {Menentukan string tertinggi berdasarkan leksikografis}
<b>KAMUS LOKAL</b> i : <u>integer</u> maks : <u>string</u>
<b>ALGORITMA</b> i <u>traversal</u> [0..length-1] <u>if</u> (array[i] > maks) <u>then</u> maks <- array[i] -> maks
<b><u>function</u> int_min</b> (array : array [0..length-1] of <u>integer</u> , length : <u>integer</u> , index_sekarang : <u>integer</u> = 0, min : <u>integer</u> = 999999 ) ->

<u>integer</u> {Mengembalikan nilai terkecil pada array integer, asumsi nilai maksimum tertinggi adalah 999999 }
<b>KAMUS LOKAL</b> min, length : <u>integer</u>
<b>ALGORITMA</b> <u>depend on</u> (array[index_sekarang],min,index_sekarang,length) array[index_sekarang] < min : min <- array[index_sekarang] index_sekarang = length-1 : -> min <u>else</u> -> int_min(array, length, index_sekarang+1, min)
<b><u>function int_maks</u></b> (array : array [0..length-1] of <u>integer</u> , length : <u>integer</u> ) -> <u>integer</u> {Mengembalikan nilai terbesar pada array integer}
<b>KAMUS LOKAL</b> maks, i : <u>integer</u>
<b>ALGORITMA</b> i <u>traversal</u> [0..length-1] <u>if</u> (array[i] > maks) <u>then</u> maks <- array[i] -> maks
<b><u>function int_join</u></b> (array1 : array [0..length-1] of <u>integer</u> , array2 : array [0..length-1] of <u>integer</u> , length : <u>integer</u> ) -> <u>integer</u> {Menjumlahkan tiap index pada dua array integer yang sama ukuran}
<b>KAMUS LOKAL</b> i : <u>integer</u> array_hasil : array of <u>integer</u>
<b>ALGORITMA</b> i <u>traversal</u> [0..length-1] array_hasil[i] <- array1[i] + array2[i] -> array_hasil
<b><u>function Data_append</u></b> (data : Data, elemen: array[0..data.n_kolom-1] of <u>string</u> )

```
function data_append(data: Data, elemen: list[str]) -> Data:  
{Program ini untuk menambahkan suatu elemen ke dalam array, Asumsi tipe array  
dan elemen pasti sama}
```

**KAMUS LOKAL**

```
i, n_baris : integer  
isi_data, hasil : matrix of string
```

**ALGORITMA**

```
isi_data <- data.isi  
n_baris <- data.n_baris  
i traversal [0..range(n_baris)]  
    hasil[i] <- isi_data[i]  
hasil[n_baris] <- elemen  
data.isi <- hasil  
data.n_baris <- data.n_baris + 1  
-> data
```

```
function data_remove(data: Data, index: int) -> Data:  
{Program ini untuk menghapuskan data pada index yang diberikan}
```

**KAMUS LOKAL**

```
i, n_baris, n_kolom : integer  
isi_data, data_baru : matrix of string
```

**ALGORITMA**

```
isi_data <- data.isi  
n_baris <- data.n_baris  
n_kolom <- data.n_kolom  
i traversal [0..range(index)]  
    data_baru[i] <- isi_data[i]  
i traversal [0..range(index+1,n_baris)]  
    data_baru[i-1] <- isi_data[i]  
data.isi <- data_baru  
data.n_baris <- data.n_baris - 1  
-> data
```

```
function cari_index_username(users: Data, username: str) -> int:  
{Program ini untuk mencari username pada data users, jika tidak ditemukan maka  
mengembalikan -1, jika ditemukan maka mengembalikan index}
```

**KAMUS LOKAL**

```
i, n_baris_users: integer  
isi_users: matrix of string
```

**ALGORITMA**

```
i traversal [0..range(n_baris_users)]
```

<pre>         <u>if</u>(username = isi_users[i][0]) <u>then</u>             -&gt; i     -&gt; -1 </pre>
<pre> <u>function</u> cari_index_candi(users: Data, jin_pembuat: str) -&gt; list[list[int],int]: {Program ini untuk mencari semua indeks candi yang dibuat oleh jin_pembuat} </pre>
<p><b>KAMUS LOKAL</b></p> <pre>     n_baris_candi,i,count_candi : <u>integer</u>     indexs : array of <u>integer</u>     isi_candi : matrix of string </pre>
<p><b>ALGORITMA</b></p> <pre>     i <u>traversal</u> [0..range(n_baris_candi)]         <u>if</u>(isi_candi[i][1] = jin_pembuat) <u>then</u>             count_candi &lt;- count_candi + 1     j &lt;- 0     i <u>traversal</u> [0..range(count_candi)]         <u>if</u>(isi_candi[i][1] = jin_pembuat) <u>then</u>             indexs[j] &lt;- i             j &lt;- j + 1     -&gt; [indexs,count_candi] </pre>
<pre> <u>function</u> matrix_str_join(array: list[str], matriks: list[list[str]],n_baris: int, n_kolom: int) -&gt; list[list[str]]: {Program ini untuk menggabungkan array of string ke dalam matrix of string} </pre>
<p><b>KAMUS LOKAL</b></p> <pre>     i : <u>integer</u>     hasil : matrix of string </pre>
<p><b>ALGORITMA</b></p> <pre>     i <u>traversal</u> [0..range(n_baris)]         hasil[i] &lt;- matriks[i]     hasil[n_baris] &lt;- array     -&gt; hasil </pre>
<pre> <u>procedure</u> candi_append(candi: Data, array: list[str]) -&gt; None: {Program ini untuk menambahkan data pembangun suatu candi ke dalam data Candi secara terurut dan mengisi index yang kosong} </pre>
<p><b>KAMUS LOKAL</b></p> <pre>     i, n_baris, n_kolom_candi, index : <u>integer</u>     isi_candi, isi_candi_baru : matrix of string </pre>
<p><b>ALGORITMA</b></p> <pre>     index &lt;- -1 </pre>



```

i traversal [0..range(n_baris_candi)]
  if(isi_candi[i][0] ≠ str(i+1)) then
    index <- i
    break
if(index = -1) then
  candi <-
    data_append(candi,[str(n_baris_candi+1),array[0],array[1],array[2],array[3]])
else
  i traversal [0..index]
    isi_candi_baru[i] <- isi_candi_baru[i]
  i traversal [index+1..n_baris_candi+1]
    isi_candi_baru[i] <- isi_candi_baru[i-1]
  isi_candi_baru[index] <-
    [str(index+1),array[0],array[1],array[2],array[3]]
  candi.isi <- isi_candi_baru
  candi.n_baris <- candi.n_baris + 1

```

```

function generate_bahan_bangunan() -> list[int]:
{Program ini untuk mengembalikan bahan bangunan yang sudah random}

```

#### KAMUS LOKAL

```

pasir, batu, air : integer

```

#### ALGORITMA

```

pasir <- randomize(1,5)
batu <- randomize(1,5)
air <- randomize(1,5)
-> [pasir, batu, air]

```

## Modul data

#### Module data

```

{ Berisi fungsi/prosedur yang dapat membuat / mengakses file}

```

#### KAMUS

```

type Data : < isi : array [0..n_baris-1] of
              array [0..n_kolom-1] of string,
              n_baris : integer,
              n_kolom : integer >

```

#### ALGORITMA

```

from tools import string_split, string_slice

```

Di bawah ini adalah fungsi dan prosedur yang terdapat di dalamnya:

**function data\_load**

(path : string) -> Data  
{Membaca data csv di lokasi "path" dan mengembalikan data dalam format [isi\_data, n\_baris, n\_kolom]}

**KAMUS LOKAL**

n\_baris, n\_kolom, i : integer  
cek\_kolom, first\_row : boolean  
file : array [0..len(file)-1] of string  
data : Data

**ALGORITMA**

```
{ Mencari jumlah efektif data }
file <- open(path, "r").readlines()
n_baris <- -1 {agar baris judul tidak ikut terbaca}
n_kolom <- 1
cek_kolom <- True
i traversal [0..len(file)-1]
  n_baris <- n_baris + 1
  if (cek_kolom) then
    cek_kolom <- False
    i traversal [0..len(line)-1]
      if(file[i] = ";") then
        n_kolom <- n_kolom + 1
{Pengambilan data, anggap data terisi dengan string kosong
""}
{ pembukaan ulang file agar dapat terbaca lagi }
file <- open(path, "r")
i <- 0
first_row <- True
i traversal [0..len(file)-1]
  if(first_row = False) then
    if(i ≠ n_baris-1) then
      data.isi[i] <- string_split(
                                string_slice(file[i], 0,
                                                len(line)-1), ";")
    else
      data.isi[i] <- string_split(line, ";")
    i <- i + 1
  else
    first_row <- False
file.close()
```

-> data

**procedure data\_save**

```
(input path : string,  
input nama_file : string,  
input data : Data)  
{Membaca data csv di lokasi "path" dan mengembalikan [data,  
jumlah_baris, jumlah_kolom], data dalam bentuk matriks}
```

**KAMUS LOKAL**

```
n_baris, n_kolom, i, j : integer  
line : string  
file : array [0..len(file)-1] of string  
isi_data : matriks of string
```

**ALGORITMA**

```
file <- open(path+"/"+nama_file+".csv","w+")  
isi_data <- data.isi  
n_baris <- data.n_baris  
n_kolom <- data.n_kolom  
{ Tulis ulang label/ baris pertama }  
if(nama_file = "user") then  
    file.write("username;password;role\n")  
else if(nama_file = "candi") then  
    file.write("id;pembuat;pasir;batu;air\n")  
else if(nama_file = "bahan_bangunan") then  
    file.write("nama;deskripsi;jumlah\n")  
i traversal [0..n_baris-1]  
    line = ""  
    j traversal [0..n_kolom-1]  
        line <- line + isi_data[i][j]  
        if(j ≠ n_kolom-1) then  
            line += ";"  
    if(i ≠ n_baris-1) then  
        line <- line + "\n"  
    file.write(line)  
file.close()
```

# Hasil Pengujian Program

Fitur	Hasil Pengujian Program
F01 - Login	<pre> &gt;&gt;&gt; login Username: Banteng Password: cintaroro Username tidak terdaftar!  &gt;&gt;&gt; login Username: Bondowoso Password: gacintaroro Password salah!  &gt;&gt;&gt; login Username: Bondowoso Password: cintaroro Selamat datang, Bondowoso! Masukkan command "help" untuk daftar command yang dapat kamu panggil </pre> <p>Gambar 3.1 - Program Login</p>
F02 - Logout	<pre> &gt;&gt;&gt; logout Logout berhasil!  &gt;&gt;&gt; logout Logout gagal! Anda belum login, silahkan login terlebih dahulu sebelum melakukan logout </pre> <p>Gambar 3.2 - Program Logout</p>
F03 - Summon Jin	<pre> &gt;&gt;&gt; summonjin Jenis jin yang dapat dipanggil: (1) Pengumpul - Bertugas mengumpulkan bahan bangunan (2) Pembangun - Bertugas membangun candi  Masukkan nomor jenis jin yang ingin dipanggil: 2  Memilih jin "Pembangun".  Masukkan username jin: asteronb Masukkan password jin: ast  Password panjangnya harus 5-25 karakter!  Masukkan password jin: asteronb  Mengumpulkan sesajen... Menyerahkan sesajen... Membacakan mantra...  Jin asteronb berhasil dipanggil!  &gt;&gt;&gt; summonjin Jenis jin yang dapat dipanggil: (1) Pengumpul - Bertugas mengumpulkan bahan bangunan (2) Pembangun - Bertugas membangun candi  Masukkan nomor jenis jin yang ingin dipanggil: 3  Tidak ada jenis jin bernomor "3"! </pre> <p>Gambar 3.3 - Program Summon Jin</p>

F04 - Hilangkan Jin	<pre>&gt;&gt;&gt; hapusjin Masukkan username jin : korin Tidak ada jin dengan username tersebut.  &gt;&gt;&gt; hapusjin Masukkan username jin : asteron Apakah anda yakin ingin menghapus jin dengan username Jin1 (Y/N)? Y  Jin telah berhasil dihapus dari alam gaib.</pre> <p>Gambar 3.4. - Program Hilangkan Jin</p>
F05 - Ubah Tipe Jin	<pre>&gt;&gt;&gt; ubahjin Masukkan username jin: korin  Tidak ada jin dengan username tersebut.  &gt;&gt;&gt; ubahjin Masukkan username jin: kulon Jin ini bertipe "Pembangun". Yakin ingin mengubah ke tipe "Pengumpul" (Y/N)? Y Jin telah berhasil diubah.</pre> <p>Gambar 3.5. - Program Ubah Tipe Jin</p>
F06 - Jin Pembangun	<pre>&gt;&gt;&gt; bangun Candi berhasil dibangun. Sisa candi yang perlu dibangun: 18  &gt;&gt;&gt; bangun # Men-generate bahan bangunan (5 pasir, 5 batu, dan 3 air) # Memiliki 9 pasir, 3 batu, 3 air Bahan bangunan tidak mencukupi. Candi tidak bisa dibangun!</pre> <p>Gambar 3.6. - Program Jin Pembangun</p>
F07 - Jin Pengumpul	<pre>&gt;&gt;&gt; kumpul Jin menemukan 5 pasir, 3 batu, dan 1 air.  &gt;&gt;&gt; kumpul Jin menemukan 3 pasir, 3 batu, dan 2 air.</pre> <p>Gambar 3.7. - Program Jin Pengumpul</p>
F08 - Batch Kumpul / Bangun	<pre>&gt;&gt;&gt; batchkumpul Mengerahkan 9 jin untuk mengumpulkan bahan. Jin menemukan total 28 pasir, 29 batu, dan 17 air.  &gt;&gt;&gt; batchkumpul Mengerahkan 9 jin untuk mengumpulkan bahan. Jin menemukan total 31 pasir, 30 batu, dan 38 air.  &gt;&gt;&gt; batchbangun Mengerahkan 7 jin untuk membangun candi dengan total bahan 26 pasir, 28 batu, dan 22 air. Jin berhasil membangun total 7 candi.  &gt;&gt;&gt; batchbangun Mengerahkan 7 jin untuk membangun candi dengan total bahan 20 pasir, 23 batu, dan 14 air. Jin berhasil membangun total 7 candi.</pre> <p>Gambar 3.8. - Program Batch Kumpul / Bangun</p>

F09 - Ambil Laporan Jin	<pre> &gt;&gt;&gt; laporanjin &gt; Total Jin: 16 &gt; Total Jin Pengumpul: 9 &gt; Total Jin Pembangun: 7 &gt; Jin Terajin: jin_kuli3 &gt; Jin Termalas: jin5 &gt; Jumlah Pasir : 30 unit &gt; Jumlah Batu : 17 unit &gt; Jumlah Air : 25 unit </pre> <p>Gambar 3.9. - Program Ambil Laporan Jin</p>
F10 - Ambil Laporan Candi	<pre> &gt;&gt;&gt; laporancandi &gt; Total Candi: 97 &gt; Total Pasir yang digunakan: 308 &gt; Total Batu yang digunakan: 287 &gt; Total Air yang digunakan: 287 &gt; ID Candi Termahal: 48 (Rp 162500) &gt; ID Candi Termurah: 18 (Rp 40000) </pre> <p>Gambar 3.10. - Program Ambil Laporan Candi</p>
F11 - Hancurkan Candi	<pre> &gt;&gt;&gt; hancurkancandi Masukkan ID candi: 910 Tidak ada candi dengan ID tersebut.  &gt;&gt;&gt; hancurkancandi Masukkan ID candi: 2 Apakah anda yakin ingin menghancurkan candi ID: 2 (Y/N)? Y Candi telah berhasil dihancurkan. </pre> <p>Gambar 3.11. - Program Hancurkan Candi</p>
F12 - Ayam Berkokok	<pre> &gt;&gt;&gt; ayamberkokok Kukuruyuk.. Kukuruyuk.. Jumlah Candi: 81  Selamat, Roro Jonggrang memenangkan permainan!  *Bandung Bondowoso angry noise* Roro Jonggrang dikutuk menjadi candi.  &gt;&gt;&gt; ayamberkokok Kukuruyuk.. Kukuruyuk.. Jumlah Candi: 100 Yah, Bandung Bondowoso memenangkan permainan! </pre> <p>Gambar 3.12. - Program Ayam Berkokok</p>

F13 - Load	<pre> PS C:\Users\Bagas\Documents\Python VS Code\Tugas-Besar-Dasar-Pemrograman-Kelompok-10-Kelas-06&gt; python main.py Tidak ada nama folder yang diberikan!  usage: main.py &lt;nama_folder&gt; PS C:\Users\Bagas\Documents\Python VS Code\Tugas-Besar-Dasar-Pemrograman-Kelompok-10-Kelas-06&gt; python main.py mba_roro Folder "save/mba_roro" tidak ditemukan. PS C:\Users\Bagas\Documents\Python VS Code\Tugas-Besar-Dasar-Pemrograman-Kelompok-10-Kelas-06&gt; python main.py tester_kel10 Loading... Selamat datang di program "Manajerial Candi"! Masukkan command "help" untuk daftar command yang dapat kamu panggil. </pre> <p>Gambar 3.13. - Program Load</p>
F14 - Save	<pre> &gt;&gt;&gt; save Masukkan nama folder: tester_kel10 Saving... Berhasil menyimpan data di folder save/tester_kel10!  &gt;&gt;&gt; save Masukkan nama folder: tester1_kel10 Saving... Membuat folder save/tester1_kel10...  Berhasil menyimpan data di folder save/tester1_kel10! </pre> <p>Gambar 3.14. - Program Save</p>
F15 - Help	<pre> &gt;&gt;&gt; help ===== HELP ===== 1. login    Untuk masuk menggunakan akun 2. save    Untuk menyimpan data permainan saat ini 3. exit    Untuk keluar dari program dan kembali ke terminal  &gt;&gt;&gt; login Username: Bondowoso Password: cintaroro Selamat datang, Bondowoso! Masukkan command "help" untuk daftar command yang dapat kamu panggil  &gt;&gt;&gt; help ===== HELP ===== 1. logout    Untuk keluar dari akun yang digunakan sekarang 2. summonjin    Untuk memanggil jin 3. hapusjin    Untuk menghapus jin 4. undohapus    Untuk mengembalikan jin yang terhapus 4. ubahjin    Untuk mengubah tipe jin (pengumpul/pembangun) 5. batchkumpul    Untuk melakukan pengumpulan oleh semua jin pengumpul 6. batchbangun    Untuk melakukan pembangunan oleh semua jin pembangun 7. laporanjin    Untuk mengambil laporan jin 8. laporancandi    Untuk mengambil laporan candi 9. save    Untuk menyimpan data permainan saat ini 10. exit    Untuk keluar dari program dan kembali ke terminal </pre>

	<pre> &gt;&gt;&gt; login Username: Roro Password: gasukabondo Selamat datang, Roro! Masukkan command "help" untuk daftar command yang dapat kamu panggil  &gt;&gt;&gt; help ===== HELP ===== 1. logout    Untuk keluar dari akun yang digunakan sekarang 2. hancurkancandi    Untuk menghancurkan candi yang tersedia 3. ayamberkokok    Untuk menyelesaikan permainan dengan memalsukan pagi hari 4. save    Untuk menyimpan data permainan saat ini 5. exit    Untuk keluar dari program dan kembali ke terminal  &gt;&gt;&gt; login Username: asteron Password: asteronwinner Selamat datang, asteron! Masukkan command "help" untuk daftar command yang dapat kamu panggil  &gt;&gt;&gt; help ===== HELP ===== 1. logout    Untuk keluar dari akun yang digunakan sekarang 2. bangun    Untuk membangun sebuah candi 3. save    Untuk menyimpan data permainan saat ini 4. exit    Untuk keluar dari program dan kembali ke terminal  &gt;&gt;&gt; login Username: jin_kuli1 Password: kulisejati Selamat datang, jin_kuli1! Masukkan command "help" untuk daftar command yang dapat kamu panggil  &gt;&gt;&gt; help ===== HELP ===== 1. logout    Untuk keluar dari akun yang digunakan sekarang 2. bangun    Untuk membangun sebuah candi 3. save    Untuk menyimpan data permainan saat ini 4. exit    Untuk keluar dari program dan kembali ke terminal </pre> <p>Gambar 3.15. - Program Help</p>
F16 - Exit	<pre> &gt;&gt;&gt; exit Apakah Anda mau melakukan penyimpanan file yang sudah diubah? (y/n) y Masukkan nama folder: tester_kel10 Saving... Berhasil menyimpan data di folder save/tester_kel10! </pre> <p>Gambar 3.16. Program Exit</p>



# Lampiran Form Asistensi

## Form MoM Asistensi Tugas Besar IF1210/Dasar Pemrograman Sem. 2 2022/2023

Nomor Asistensi : 1  
No. Kelompok/Kelas : 10/K06  
Tanggal asistensi : 8 April 2023

Anggota kelompok		NIM / Nama (Hanya yang Hadir)
1		16522024 / Bagus Wara
2		19622024 / Serenada Cinta Sunindya
3		19622078 / Roken Francisco T.B.
4		19622150 / Dedy Hofrenida Soragih
5		
6		
Asisten pembimbing		NIM / Nama
		135210117 / Hafiz Nur Rahman Ghazali

### Catatan Asistensi:

Rangkuman Diskusi
<p>Asistensi pertama membahas tentang progres tugas besar sejauh ini. Pembagian tugas belum benar-benar ditetapkan, tetapi sudah bagian dari proyek sudah dikerjakan dan Francisco yang sudah mencoba melakukan coding untuk beberapa bagian dari proyek. Dalam asistensi ini dibahas fungsi-fungsi yang boleh dan tidak boleh digunakan di dalam kode.</p> <p>Fungsi yang tidak dapat digunakan sudah tertera di FDR, tetapi diskusi kembali di asistensi ini. Fungsi seperti split, append, pop tidak boleh digunakan. Sejak ini, kode yang sudah selesai dibuat sudah mencapai FDR tetapi kode masih mengandung fungsi yang tidak seharusnya digunakan.</p> <p>Laporan sudah mulai dibuat di Google Docs. Dalam laporan tersebut, kami terdapat cover dan halaman pengantar. Progres kode sejauh ini belum ditulis di dalam laporan untuk selanjutnya.</p> <p>Saran yang diberikan:</p> <ul style="list-style-type: none"> <li>Partisipasi semua orang dalam kelompok berkontribusi dan selalu mengikuti perkembangan dari asistensi.</li> <li>Selalu laksanakan spesifikasi pedoman dan FDR untuk tugas besar.</li> </ul>
Tindak Lanjut
<p>Membuat pertemuan untuk melakukan pembagian tugas, melanjutkan pengerjaan tugas, menambahkan komentar yang sesuai dan jelas pada kode yang dibuat.</p>

08/04/2023



HAFID2 NOR R.G.

**Form MoM Asistensi Tugas Besar  
IF1210/Dasar Pemrograman  
Sem. 2 2022/2023**

Nomor Asistensi : 2  
No. Kelompok/Kelas : 10 / K06  
Tanggal asistensi : 16 April 2023

Anggota kelompok	NIM / Nama (Hanya yang Hadir)	
	1	16522024 / Bagaswara
	2	19622024 / Serenada Cinta Sunindyo
	3	19622078 / Dedy Hofmanindo Saragih
	4	19622150 / Raden Francisco Trianto Bratadiningrat
	5	
Asisten pembimbing	NIM / Nama	
	13520117 / Hafidz Nur Rahman Ghozali	

**Catatan Asistensi:**

Rangkuman Diskusi
<p>Memperlihatkan progress dari tugas besar: F01, F02, F04, F06, F08, F09, F12, F13, F14, F16, B01</p> <p>Menanyakan tentang F09 laporan jin bagian jin termalas dan terajin, kesimpulan melihat jin dari array user's saja</p> <p>Menanyakan tentang B01 algoritma random number generator, kesimpulan kita harus mengerti algoritma tersebut dulu</p> <p>Menanyakan tentang B02 tentang fungsi rekursif, kesimpulan setiap orang harus membuat fungsi rekursif ( jumlah total 4 fungsi) yang digunakan pada program serta bebas dimananya</p> <p>Menanyakan tentang B03 tentang tpying, kesimpulan hanya input dan output yang diberi typing, tapi kamus lokal tetap ada buat variabel dalam fungsi</p>
Tindak Lanjut
<p>Melanjutkan progress tugas besar dengan melengkapi dan mengoreksi program</p>

16/04/2023

Hafidz Nur R. G.