

# **2048! The game you couldn't put down**

Disusun sebagai syarat penilaian mata kuliah  
Pemrograman Python



## **Disusun Oleh:**

Septian Dwi Atmoko  
Faza Ridha Alfana

19.83.0361  
19.83.0376

Ketua  
Anggota

**Program Studi Teknik Komputer**

**Fakultas Ilmu Komputer**

**Universitas Amikom Yogyakarta**

**2020**

# Daftar Isi

HALAMAN SAMPUL.....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Manfaat.....	1
BAB II Perancangan.....	2
2.1 Team.....	2
2.2 Alat Dan Bahan.....	Error!
Bookmark not defined.....	2
BAB III Dokumentasi.....	4

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Assalamualaikum Wr.Wb Puji syukur kehadirat Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga saya dapat menyelesaikan tugas makalah untuk UAS yang berjudul 2048! The game you couldn't put down ini tepat pada waktunya.

Adapun tujuan dari penulisan dari makalah ini adalah untuk memenuhi tugas pada **Pemrograman Python** Selain itu, makalah ini juga bertujuan untuk menambah wawasan tentang **Pembuatan Game** dengan menggunakan Prgroman Python.

Kami membuat game ini karena kita ingin membuat game yang ringan dan mudah untuk dimainkan, karena situasi yang sedang pandemic seperti ini mengharuskan untuk dirumah saja, akhirnya kami memutuskan membuat game. dan sangat mirip dengan aplikasi Threes! yang dirilis sebulan sebelumnya. Cirulli membuat permainan ini dalam satu akhir pekan saja

saat mengetahui permainan buatannya dikunjungi 4 juta pengguna dalam kurang dari satu minggu. karena 2048 buatannya direncanakan sebagai proyek sampingan. Cirulli mengaku proyek tersebut "adalah caranya menghabiskan waktu". Permainan ini gratis, dan Cirulli tidak mau meraup untung dari "sesuatu yang tidak ia temukan". Dan merilis aplikasi gratisnya untuk iOS dan Android pada bulan Mei 2014.

### **1.2 Manfaat**

Kami membuat game yang di bilang di atas tadi ada Pemain baru menang ketika sebuah kotak bernomor 2048 muncul di papan. Setelah mencapai skor target, pemain bisa terus bermain dalam mode kotak pasir melewati angka 2048

Saat pemain tidak bisa bergerak lagi (tidak ada ruang kosong dan kotak bernomor sama di sekitarnya), permainan berakhir.

Cara bermainnya yang sederhana (hanya empat arah) membuat 2048 dipakai dalam video promo ikat tangan pengendali berbasis gerak tubuh buatan Myo. Kode sumber 2048 telah dimanfaatkan sebagai alat bantu bagi pengajar pemrograman. Juara kedua kontes koding di Matlab Central Exchange adalah sistem kecerdasan buatan yang mampu memainkan 2048 sendirian.

## BAB II

### PERANCANGAN

#### 2.1. Team

Pada Bagian Team, tuliskan nama lengkap dan no mahasiswa dari seluruh personil kelompok final project anda. Tuliskan juga peran dari nama tersebut lalu berikan penjelasan dari apa yang dilakukan personal tersebut berdasarkan peran yang diberikan.

Contoh:

**Nama Lengkap** : Septian Dwi Atmoko

**Peran** : Ketua Kelompok Final Project

**Uraian Pekerjaan:**

Sebagai ketua kelompok saya berperan dalam membuat karakter yang ada di game Shooter cara nya dengan menggunakan software Pycharm, IDLE Python 3.8 dan Python 3.7 gunanya hanya meng krop gambar saja yang semula satu gambar dijadikan satu persatudengan ukuran kotak 16x16 dengan format JPG tujuan nya agar bisa terbaca oleh program gamenya.

**Nama Lengkap** : Faza Ridha Alfana

**Id Github** : siputterbang

**Peran** : Anggota

**Uraian Pekerjaan:**

Sebagai anggota, saya mengambil bagian dalam pembuatan game 2048 serta membuat main menu atau tampilan awal game tersebut. Dalam pembuatan game ini saya menggunakan software pycharm.

#### 2.2. Alat dan Bahan

Pada bagian alat dan bahan, perlu dituliskan spesifikasi software (framework / library) yang digunakan dalam penyusunan final project.

No	Package Name	Version	Installation
Major Package			
		3.8.4	On Linux: Apt-get install python3.8  On Windows 64bit: <a href="https://www.python.org/ftp/python/3.8.2/">https://www.python.org/ftp/python/3.8.2/</a>

1.	Python 3	dan 3.7.9	python-3.8.0-amd64.exe  On Windows 32bit: <a href="https://www.python.org/ftp/python/3.8.2/python-3.8.0.exe">https://www.python.org/ftp/python/3.8.2/python-3.8.0.exe</a>
2.	Pygame		Di linux & Windows Pip install pygame
3.	Random		Sudah ada dalam package python

## BAB III

### DOKUMENTASI

Dalam dokumentasi ini kami akan menjelaskan proses mengenai alur game yang kami buat beserta fungsi atau modul, kami membuat

#### 1. Modul

Kami menggunakan 3 modul dalam game yang kami buat, modul tersebut meliputi pygame dan tkinter sebagai modul utama untuk membuat game serta modul random dan colors sebagai modul tambahan.

```
import tkinter as tk
import random
import colors as c
```

#### 2. Font

Untuk menampilkan tulisan pada game dapat menggunakan pygame.font dan untuk memutar musik dapat menggunakan pygame.mixer, sebelum kedua perintah ini digunakan maka harus menginisialisasi terlebih dahulu di awal program dengan menuliskan python.frame.init() dan pygame.init().

```
class Game(tk.Frame):
    def __init__(self):
        tk.Frame.__init__(self)
        self.grid()
        self.master.title('2048')
```

#### 3. Layar

Kami menggunakan resolusi 720,720 dan cara untuk menampilkan GUI window nya kita menggunakan pygame.display.set\_mode. dan pygame.display.set\_caption untuk menampilkan nama game di pojok kiri atas.

```

self.main_grid = tk.Frame(
    self, bg=c.GRID_COLOR, bd=3, width=400, height=400)
self.main_grid.grid(pady=(80, 0))
self.make_GUI()
self.start_game()

```

#### 4. Start Player

Pada bagian ini untuk menampilkan file gambar angka kecil hingga besar ke program dengan cara `pygame.image.load`, musuh yang kami buat ada 16 kotak dalam angka tersebut.

```

def start_game(self):
    # create matrix of zeroes
    self.matrix = [[0] * 4 for _ in range(4)]

    # fill 2 random cells with 2s
    row = random.randint(0, 3)
    col = random.randint(0, 3)
    self.matrix[row][col] = 2
    self.cells[row][col]["frame"].configure(bg=c.CELL_COLORS[2])
    self.cells[row][col]["number"].configure(
        bg=c.CELL_COLORS[2],
        fg=c.CELL_NUMBER_COLORS[2],
        font=c.CELL_NUMBER_FONTS[2],
        text="2")
    while(self.matrix[row][col] != 0):
        row = random.randint(0, 3)
        col = random.randint(0, 3)
    self.matrix[row][col] = 2
    self.cells[row][col]["frame"].configure(bg=c.CELL_COLORS[2])
    self.cells[row][col]["number"].configure(
        bg=c.CELL_COLORS[2],
        fg=c.CELL_NUMBER_COLORS[2],
        font=c.CELL_NUMBER_FONTS[2],
        text="2")

```

Kami menggunakan software PyCharm Gambar (1) untuk mengcrop foto Gambar (2) kami crop satu persatu mejadi ukuran 64x64 dan dijadikan file JPG untuk bisa dibaca oleh program.

Cara bermain:

Tombol panah atau tandukan pengguna pada keyboard Anda untuk memindahkan ubin. Gabungkan mereka untuk membuat ubin dengan angka 2048. Setiap giliran, ubin baru akan muncul secara acak di tempat kosong di papan dengan nilai 2 atau 4.

Gambar (1)



Gambar (2)





##### 5. Player Utama Pada GUI.

Player utama memiliki 5 cells dari berbagai ukuran.

Fungsi didefinisikan sebagai fungsi lambda yang mengambil contoh Game, yang mencakup semua efek khusus dalam Breakout yang mungkin ingin diubah. Lalu seperti apa sih tampilan GUI yang akan di hasilkan setelah kita jalankan program diatas , Saya akan tampilkan contoh GUI screenshotnya di Bawah ini. Ketika pertama kali window tersebut muncul, maka tidak ada tulisan seperti yang terlihat digambar tersebut

```
def make_GUI(self):
    # make grid
    self.cells = []
    for i in range(4):
        row = []
        for j in range(4):
            cell_frame = tk.Frame(
                self.main_grid,
                bg=c.EMPTY_CELL_COLOR,
                width=100,
                height=100)
            cell_frame.grid(row=i, column=j, padx=5, pady=5)
            cell_number = tk.Label(self.main_grid, bg=c.EMPTY_CELL_COLOR)
            cell_number.grid(row=i, column=j)
            cell_data = {"frame": cell_frame, "number": cell_number}
            row.append(cell_data)
        self.cells.append(row)
```

## 6. Background Layar

Pada bagian background,terdapat 2 nama background yaitu backgrnd sebagai background game dan background awal sebagai background main menu.

Transfrom scale adalah membuat gambar background full layar meskipun dengan ukuran gambar terbatas.

```
GRID_COLOR = "#a39489"
EMPTY_CELL_COLOR = "#c2b3a9"
SCORE_LABEL_FONT = ("Verdana", 20)
SCORE_FONT = ("Helvetica", 32, "bold")
GAME_OVER_FONT = ("Helvetica", 48, "bold")
GAME_OVER_FONT_COLOR = "#ffffff"
WINNER_BG = "#ffcc00"
LOSER_BG = "#a39489"
```

## 7. Cells Colors

Ada beberapa warna yang kami gunakan,yaitu pada saat main menu dan saat game dimulai dari berbagai kode bervariasi cells nya.

```
CELL_COLORS = {
    2: "#fcefe6",
    4: "#f2e8cb",
    8: "#f5b682",
    16: "#f29446",
    32: "#ff775c",
    64: "#e64c2e",
    128: "#ede291",
    256: "#fce130",
    512: "#ffdb4a",
    1024: "#f0b922",
    2048: "#fad74d"
}
```

Untuk menambahkan warna cukup membuat angka dan mengisinya dengan perintah pycharm.(namafile). Untuk memutar musik cukup dengan perintah play tekan tombol run dan stop dengan cara menghentikannya tekan tombol close.

Ada cell number color dan cell number fonts berfungsi sebagai

```
CELL_NUMBER_COLORS = {
    2: "#695c57",
    4: "#695c57",
    8: "#ffffff",
    16: "#ffffff",
    32: "#ffffff",
    64: "#ffffff",
    128: "#ffffff",
    256: "#ffffff",
    512: "#ffffff",
    1024: "#ffffff",
    2048: "#ffffff"
}
```

```
CELL_NUMBER_FONTS = {
    2: ("Helvetica", 55, "bold"),
    4: ("Helvetica", 55, "bold"),
    8: ("Helvetica", 55, "bold"),
    16: ("Helvetica", 50, "bold"),
    32: ("Helvetica", 50, "bold"),
    64: ("Helvetica", 50, "bold"),
    128: ("Helvetica", 45, "bold"),
    256: ("Helvetica", 45, "bold"),
    512: ("Helvetica", 45, "bold"),
    1024: ("Helvetica", 40, "bold"),
    2048: ("Helvetica", 40, "bold")
}
```

## 8. Class Matrix

Class ini digunakan sebagai template yang nantinya saat memainkan angka tinggal menggunakan class ini sehingga tidak perlu mendefinisikan cara memainkan hanya memakai keyboard pada tiap-tiap angka player yang jumlahnya sangat banyak. Dengan matrix manipulasi fungsi def stack dan def combine.

*# Matrix Manipulation Functions*

```
def stack(self):
    new_matrix = [[0] * 4 for _ in range(4)]
    for i in range(4):
        fill_position = 0
        for j in range(4):
            if self.matrix[i][j] != 0:
                new_matrix[i][fill_position] = self.matrix[i][j]
                fill_position += 1
    self.matrix = new_matrix

def combine(self):
    for i in range(4):
        for j in range(3):
            if self.matrix[i][j] != 0 and self.matrix[i][j] == self.matrix[i][j + 1]:
                self.matrix[i][j] *= 2
                self.matrix[i][j + 1] = 0
                self.score += self.matrix[i][j]
```

## 9. Class 2048 game

```
def reverse(self):
    new_matrix = []
    for i in range(4):
        new_matrix.append([])
        for j in range(4):
            new_matrix[i].append(self.matrix[i][3 - j])
    self.matrix = new_matrix

def transpose(self):
    new_matrix = [[0] * 4 for _ in range(4)]
    for i in range(4):
        for j in range(4):
            new_matrix[i][j] = self.matrix[j][i]
    self.matrix = new_matrix
```

Class ini juga digunakan sebagai template yang nantinya saat akan menampilkan pesawat musuh maupun pesawat player

Fungsi dari variabel :

Fungsi dari **def reverse** = yang berfungsi untuk mengembalikan iterator yang berisi string terbalik. Satu persatu karakter dari string yang sudah dibalik kemudian digabungkan menggunakan metode join() string

Fungsi dari **def transpose** = yang berfungsi untuk proses mengubah baris ke – 1 menjadi kolom ke – 1, baris ke – 2 menjadi kolom ke – 2, baris ke – 3 menjadi kolom ke – 3 yang lain.

Fungsi dari **def add new title** = yang berfungsi menampilkan menu-menu atau fitur yang ada di dalam aplikasi.

Fungsi dari **def update GUI** = yang berfungsi memperbarui satu atau beberapa nilai di dalam list dengan memberikan potongan di sisi kiri operator penugasan

Fungsi dari **def up** = yang berfungsi menaikkan postingan atau memunculkan kembali postingan tersebut ke atas

Fungsi dari **def down** = yang berfungsi untuk menurunkan postingan tersebut ke bawah

Fungsi dari **def right** = yang berfungsi untuk membelokkan ke kanan dalam permainan

Fungsi dari **def left** = yang berfungsi untuk membelokkan ke kiri ke dalam permainan dalam tombolnya.

Fungsi dari **def game over** = yang berfungsi untuk mengakhiri permainan tombol pengeseran angka

## 10. Class Player tile random (2048)

```
# Add a new 2 or 4 tile randomly to an empty cell

def add_new_tile(self):
    row = random.randint(0, 3)
    col = random.randint(0, 3)
    while(self.matrix[row][col] != 0):
        row = random.randint(0, 3)
        col = random.randint(0, 3)
    self.matrix[row][col] = random.choice([2, 4])
```

Class ini digunakan untuk siplayer, pada class ini nilai/value dari class pesawat diturunkan ke dalam class Player. Ada beberapa tambahan fungsi yaitu poin untuk menampilkan point dalam game serta untuk menampilkan banyaknya dimiliki siplayer

## 11. Class Update the GUI

```
# Update the GUI to match the matrix
```

```
def update_GUI(self):  
    for i in range(4):  
        for j in range(4):  
            cell_value = self.matrix[i][j]  
            if cell_value == 0:  
                self.cells[i][j]["frame"].configure(bg=c.EMPTY_CELL_COLOR)  
                self.cells[i][j]["number"].configure(  
                    bg=c.EMPTY_CELL_COLOR, text="")  
            else:  
                self.cells[i][j]["frame"].configure(  
                    bg=c.CELL_COLORS[cell_value])  
                self.cells[i][j]["number"].configure(  
                    bg=c.CELL_COLORS[cell_value],  
                    fg=c.CELL_NUMBER_COLORS[cell_value],  
                    font=c.CELL_NUMBER_FONTS[cell_value],  
                    text=str(cell_value))  
        self.score_label.configure(text=self.score)  
        self.update_idletasks()
```

Manfaat dari class ini sama seperti class Player, terdapat variabel dict yang nantinya digunakan sebagai gambar dan angka yang digunakan tombol angka.. Fungsi **bergerak** digunakan untuk menggerakan angka secara otomatis dari atas layar ke bawah layar dan ke atas layar

```
def up(self, event):
    self.transpose()
    self.stack()
    self.combine()
    self.stack()
    self.transpose()
    self.add_new_tile()
    self.update_GUI()
    self.game_over()
```

```
def down(self, event):
    self.transpose()
    self.reverse()
    self.stack()
    self.combine()
    self.stack()
    self.reverse()
    self.transpose()
    self.add_new_tile()
    self.update_GUI()
    self.game_over()
```

Pada class ini digunakan apabila player namun ada yang berhasil tertentu akan menambah jumlah angka yang dimiliki si 2048.

**def** digunakan untuk mendeteksi apabila check if any moves are possible, saling satuan sama lain Alur angka pada game ini saat pertamakali dimulai adalah membuka main menu.



```
# Check if any moves are possible

def horizontal_move_exists(self):
    for i in range(4):
        for j in range(3):
            if self.matrix[i][j] == self.matrix[i][j + 1]:
                return True
    return False

def vertical_move_exists(self):
    for i in range(3):
        for j in range(4):
            if self.matrix[i][j] == self.matrix[i + 1][j]:
                return True
    return False
```

Kemudian membuat variabel *player* yang diisi dengan class **Player** lokasi ini adalah lokasi awal player, apabila player mengalami kekalahan maka kemudian menampilkan tulisan *Game Over* lalu

```
# Check if Game is Over (Win/Lose)

def game_over(self):
    if any(2048 in row for row in self.matrix):
        game_over_frame = tk.Frame(self.main_grid, borderwidth=2)
        game_over_frame.place(relx=0.5, rely=0.5, anchor="center")
        tk.Label(
            game_over_frame,
            text="You win!",
            bg=c.WINNER_BG,
            fg=c.GAME_OVER_FONT_COLOR,
            font=c.GAME_OVER_FONT).pack()
```

Dibawah **def** ada perulangan *for* lagi dimana perulangan ini akan secara acak menampilkan angka yang jika player menabraknya akan menambah nyawa siplayer.

```

elif not any(0 in row for row in self.matrix) and not self.horizontal_move_exists() and not self.vertical_move_exists():
    game_over_frame = tk.Frame(self.main_grid, borderwidth=2)
    game_over_frame.place(relx=0.5, rely=0.5, anchor="center")
    tk.Label(
        game_over_frame,
        text="Game over!",
        bg=c.LOSER_BG,
        fg=c.GAME_OVER_FONT_COLOR,
        font=c.GAME_OVER_FONT).pack()

```

Kemudian dibawahnya lagi sebelum perulangan *while* ada *player.draw* untuk menampilkan gambar player kelayar dan *pygame*. *Def main* untuk mengakhiri game main layar.

```

def main():
    Game()

if __name__ == "__main__":
    main()

```

Ketika pertama kali window tersebut muncul, maka tidak ada tulisan seperti yang terlihat digambar tersebut

Dan yang terakhir saya ucapkan terimakasih pada kalian kurang lebihnya saya mohon maaf dan mohon di maklumi ya.. karena saya juga masih belajar dan tentu saja tidak luput dari rasa salah dan khilaf.. oke.. sekian dulu ceramah nya..

THE END !!!