APLIKASI GAMES 8-PUZZLE

“PAK JEUL”

Dokumen Spesifikasi Program “Pak Jeul” untuk MK Proyek Perangkat Lunak 2



Disusun Oleh :

KELOMPOK 3

1. Ananda Khoirunisa (171511003)
2. Kiki Pratiwi (171511046)
3. Mufida Nuha Salimah (171511050)
4. Nikita Nabila (171511054)
5. Velia Sagita Putri (171511065)

Manager

Ida Suhartini

**Jurusan Teknik Komputer dan Informatika**

**Program Studi D3 Teknik Informatika**

**Politeknik Negeri Bandung**

**2018**

FORM PENILAIAN UTS PROYEK 2

|  |  |  |
| --- | --- | --- |
| Kelompok | 3 | Manajer |
| Judul | Aplikasi Games 8-Puzzle “Pak Jeul” | Ida Suhartini |

1. PENILAIAN INDIVIDU

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | NIM | Nama | Tanda  Tangan | Kontribusi laporan (%) | Kontribusi Aplikasi (%) | Kontribusi  Presentasi (%) |
| 1. | 171511003 | Ananda Khoirunisa |  |  |  |  |
| 2. | 171511046 | Kiki Pratiwi |  |  |  |  |
| 3. | 171511050 | Mufida Nuha Salimah |  |  |  |  |
| 4. | 171511054 | Nikita Nabila |  |  |  |  |
| 5. | 171511066 | Velia Sagita Putri |  |  |  |  |

1. PENILAIAN MANAGER

LAPORAN

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Spesifikasi | | Perancangan | | Lesson  Learn | Kesimpulan  dan Saran | Total |
| Deskripsi Fitur | Deskripsi Gameplay | Struktur Data | Desain GUI |
|  |  |  |  |  |  |  |

APLIKASI

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Struktur Data | | Komponen Game | | Total |
| Array Inisial | Array Goal | Gameplay | Estetika |
|  |  |  |  |  |

PRESENTASI

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Presentasi | | Tanya Jawab | | Coding | |
| Slide | Pemaparan | Penguasaan  Bahan | Kesesuaian Dokumen | Modularity | Clean Code |
| 1. |  |  |  |  |  |  |
| 2. |  |  |  |  |  |  |
| 3. |  |  |  |  |  |  |
| 4. |  |  |  |  |  |  |
| 5. |  |  |  |  |  |  |

# **KATA PENGANTAR**

Puji syukur kami panjatkan ke hadirat Allah SWT yang telah memberikan rahmat serta karunia-Nya sehingga penyusunan laporan progress kerja dalam pembuatan Puzzle Game dapat terselesaikan.

Laporan ini berisi tentang progress kerja selama menjalankan proyek 2 dari pertemuan pertama hingga pertemuan keenam. Diharapkan dengan laporan ini dapat memberikan informasi yang lengkap dan jelas mengenai bagaimana proses pembuatan Puzzle Game selama proyek 2 ini. Dari segi laporan, presentasi, kinerja serta aspek lainnya selama ini masih jauh dari kata sempurna, oleh karena itu kritik, saran, dan bimbingan dari semua pihak yang bersifat membangun sangat harapkan demi menciptakan hasil yang lebih baik dikesempatan berikutnya.

Akhir kata, terima kasih kepada semua pihak yang telah ikut berperan dalam pembuatan laporan ini dari awal sampai akhir, semoga Allah SWT senantiasa meridhai segala usaha dan kinerja kita. Semoga laporan ini dapat bermanfaat bagi para pembaca. Aamiin

|  |
| --- |
| Bandung, Juni 2018  Penyusun |
|  |

# **DAFTAR ISI**

[**KATA PENGANTAR** i](#_Toc509951200)

[**DAFTAR ISI** ii](#_Toc509951201)

[**DAFTAR GAMBAR** iii](#_Toc509951202)

[**DAFTAR TABEL** iii](#_Toc509951203)

[**BAB I** 1](#_Toc509951204)

[**1.** **Latar Belakang** 1](#_Toc509951205)

[**2.** **Rumusan Masalah** 1](#_Toc509951206)

[**3.** **Tujuan dan Manfaat** 1](#_Toc509951207)

[**BAB II** 3](#_Toc509951208)

[**1.** **Deskripsi Program** 3](#_Toc509951209)

[**a.** **Identifikasi Proses** 3](#_Toc509951210)

[**b.** **Identifikasi Data** 4](#_Toc509951211)

[**c.** **Perancangan Struktur Menu** 21](#_Toc509951212)

[**d.** **Desain GUI** 23](#_Toc509951213)

[**BAB III** 28](#_Toc509951214)

[**1.** **Lesson Learn** 28](#_Toc509951215)

[**BAB IV** 30](#_Toc509951216)

[**1.** **Kesimpulan** 30](#_Toc509951217)

[**Progress Kerja** 31](#_Toc509951218)

[**Daftar Pustaka** 41](#_Toc509951219)

# **DAFTAR GAMBAR**

Gambar 1 : Struktur splashscreen

Gambar 2 : Struktur main menu

Gambar 3 : Struktur setgoal and level & play

Gambar 4 : Struktur rank

Gambar 5 : GUI splashscreen

Gambar 6 : GUI main menu

Gambar 7 : GUI set goal and level

Gambar 8 : GUI playgame screen

Gambar 9 : GUI result screen

Gambar 10 : GUI about screen

Gambar 11 : GUI help screen

Gambar 12 : GUI rank screen

Gambar 13 : GUI input name screen

# **DAFTAR TABEL**

Tabel 1 : Konstanta

Tabel 2 : Kamus data global

Tabel 3 : Modul

# **BAB I**

**PENDAHULUAN**

## **Latar Belakang**

8 puzzle adalah permainan sliding puzzle ciptaan Sam Loyd yang terdiri dari sebuah bingkai yang berisi kotak kotak angka dalam keadaan teracak dimana sebuah kotak hilang. Karena permainan ini bernama 8 puzzle, maka ukuran bingkai adalah 3 x 3. tujuan dari puzzle ini adalah dengan memakai gerakan geser yang menggunakan ubin kosong untuk mencapai goal state.

8 puzzle merupakan permainan dengan genre board and puzzle games berupa menyusun ulang kepingan suatu gambar atau angka supaya membentuk suatu susunan yang diinginkan. Pada awalnya puzzle dibuat untuk membantu dalam pelajaran geografi bagi anak-anak sekolah pada tahun 1766 oleh John Spilsbury. Ditemukan dan dipopulerkan oleh Noyes Palmer Chapman pada 1870-an. Ini dimainkan pada grid 3-oleh-3 dengan 8 blok persegi berlabel 1 hingga 8 dan kotak kosong. Tujuannya adalah mengatur ulang blok-blok agar sesuai urutan. Anda diizinkan untuk menggeser blok secara horizontal atau vertikal ke dalam kotak kosong. Berikut ini menunjukkan urutan langkah hukum dari posisi papan awal (kiri) ke posisi tujuan (kanan).. Sejak saat itu hingga saat ini, mulai bermunculan jenis puzzle terbaru. Dengan variasi dan tingkat kesulitan yang bermacam-macam.

Yang dibuat kali ini adalah 8 puzzle dengan menggunakan Bahasa C dengan IDE-nya yaitu Dev-C++. Game 8 puzzle ini dibuat dengan bantuan header graphics.h

## **Rumusan Masalah**

Dari latar belakang tersebut,dapat disimpulkan rumusan masalahnya adalah sebagai berikut :

1. Bagaimana memrogram sebuah program visual menggunakan Bahasa yang dasarnya adalah teks ?
2. Bagaimana caranya membuat game 8 puzzle dengan menggunakan Bahasa C ?
3. Bagaimana menerapkan ADT tree pada game 8 puzzle ?

## **Tujuan dan Manfaat**

Adapun tujuan dan manfaat dari kegiatan ini adalah :

1. Dapat membuat objek grafis dengan Bahasa C
2. Dapat meengintegrasikan program berbasis teks dengan yang berbasis objek
3. Dapat membuat game 8 puzzle dengan menggunakan Bahasa C.
4. Dapat mengimplementasikan ADT tree dalam games 8 puzzle.

# **BAB II**

**ISI**

## **Deskripsi Program**

Pak Jeul adalah permainan puzzle 8 yang merupakan puzzle geser yang terdiri dari bingkai ubin persegi dengan nomor acak dengan satu ubin yang hilang. Objek dari teka – teki ini adalah menempatkan ubin sesuai dengan target dengan membuat gerakan yang menggunakan ruang kosong.

### Identifikasi Proses

Proses – proses utama yang terdapat pada program adalah sebagai berikut :

1. Tahap Persiapan

Proses pemilihan menu yang akan dilanjutkan ke UI lain.

1. Start game : memulai permainan 8 puzzles yang sebelumnya akan dipanggil User Iinterface (UI) set level & goal sebelum akhirnya menuju UI play
2. About : berisi penjelasan singkat dari game dan pembuat game.
3. Rank ; menampilkan urutan rekor waktu tertinggi yang diraih user dari masing-masing level
4. Help : berisi instruksi cara memainkan games.
5. Exit : keluar dari aplikasi game.
6. Tahap Pemilihan goal dan level /setting goal and level.

Proses untuk memilih mode permainan sebelum memulai permainan

1. Memilih goal permainan, disediakan tiga pilihan goal yang bebas dipilih oleh pemain. Dua pilihan goal statis dan satu custom goal yang dipilih sesuai keinginan pemain.
2. Memilih level permainan, dibedakan berdasarkan waktu yang disediakan untuk menyelesaikan puzzle
3. level easy = 10 menit
4. level medium = 5 menit
5. level hard = 3 menit
6. Tahap Permainan

Permainan ini merupakan penyelesaian puzzle dengan menggeser kotak kosong agar puzzle terselesaikan sesuai dengan goal yang telah dipilih oleh user selama waktu yang disesuaikan dengan pilihan level.

**Aturan permainan :**

1. Pemain memilih tiga setting goal atau susunan yang diminta untuk menyelesaikan 8 Puzzles, dan memilih salah satu level yaitu easy, medium dan hard, dimana setiap level memiliki time limit / batas waktu yang berbeda untuk menyelesaikan 8 Puzzles.
2. Pemain dapat menggerakkan kotak kosong agar puzzle sesuai dengan setting goal yang telah dipilih sebelumnya.
3. Jika waktu telah habis, maka pemain dinyatakan gagal menyelesaikan 8 Puzzles.

### Identifikasi Data

* **Konstanta**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Konstanta** | **Nilai** | **Fungsi** |
| 1 | Minimum | 25 | Batas minimum board puzzle |
| 2 | Maksimum | 475 | Batas maximum board puzzel |
| 3 | Width | 900 | Ukuran lebar window |
| 4 | Height | 600 | Ukuran tinggi window |
| 5 | Margin\_Left | 220 | Margin agar window tepat ditegah layar |
| 6 | Margin\_Right | 50 | Margin agar window tepat ditegah layar |
| 7 | nPzl | 9 | Jumlah elemen array puzzle |
|  | Original Point | 0 |  |
|  | N | 100 |  |
|  | Bottom | 3 | Perpindahan element array pada saat puzzle bergerah ke bawah |
|  | Left | -1 | Perpindahan element array pada saat puzzle bergerah ke kiri |
|  | Right | 1 | Perpindahan element array pada saat puzzle bergerah ke kanan |
|  | Top | -3 | Perpindahan element array pada saat puzzle bergerah ke atas |
|  | Limit\_A | 80 |  |
|  | Limit\_B | 100 |  |
|  | Err\_No\_Num | -1 |  |
|  | Err\_No\_Mem | -2 |  |

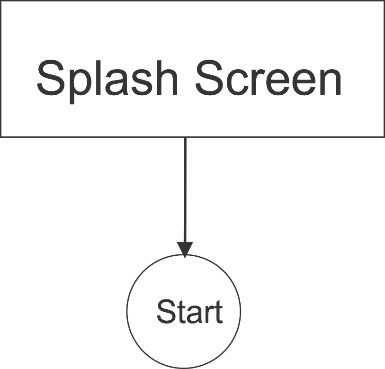
* **Kamus Data Global**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Sturktur data** | **Nama sub variabel** | **Jenis data** | **Fungsi** |
| 1 | Point | x | integer | Menunjukkan koordinat titik pada sumbu x |
| y | integer | Menunjukkan koordinat titik pada sumbu y |
| 2 | Times | second | integer | Menunjukkan detik waktu |
| minutes | innteger | Menunjukkan menit waktu |
| 3 | Game\_Setting | Goal | Character | Menunjukkan goal yang dipilih user |
| Level | Pointer of character | Menunjukkan level yang dipilih user |
| Time\_Limit | Times | menunjukkan waktu permainan sesuai level yang dipilih |
| Song | Pointer of character | Menentukan lagu yang dimainkan pada UI tertentu |
| ArrGoal[nPzl] | Array of character | Menunjukkan goals puzzle dalam array |
| 4 | Player | Name | Array of character dengan 7 elemen | Menghimpun character nama player |
| Level | Array of character dengan 7 elemen | Menunjukkan level yang dipilih |
| Movement | Integer | Menunjukkan banyaknya pergerakan yang dilakukan |
| Time\_left | Times | Menunjukkan waktu yang tersisa saat puzzle terselesaikan |
| result | Pointer character | Status games saat waktu berakhir apakan failed atau success |
| Win | Boolean | Status kemenangan games apakah berhasil menyelesaikan permainan atau tidak |
| 5 | infoMovement | status | Array of Boolean dengan 4 elemen |  |
| 6 | nodeTree |  | Pointer of address | Menjadi penunjuk pada tree |
| 7 | node |  | Pointer of AddressLink | Menjadi penunjuk pada stack |
| 8 | infotypeLink |  | Address | Menjadi penunjuk info node pada tree yang akan dimasukkan kedalam stack |
| 9 | infotype | Move | Short integer | Elemen info yang berisi perpindahan dari elemen array |
| Reverse | Short integer | Elemen info yang berisi pengembalian posisi dari elemen array menuju goal sebagai kebalikan dari move |
| Empty | Short integer | Info yang berisi posisi array kosong yang bisa dipindahkan |
| 10 | node | info | infotypeLink | Merupakan pointer yang menunjuk node pada tree yang masuk ke stack |
| next | addressLink | Merupakan pointer untuk menunjuk node selanjutnya pada stack |
| 11 | nodeTree | infoTree | infotype | Berisi info pergerakan pada tree |
| parent | address | Merupakan pointer yang menunjuk ke parent dari node |
| firstSon | address | Merupakan pointer yang menunjuk ke anak pertama dari node |
| nextBrother | address | Merupakan pointer yang menunjuk ke sibling dari first son dari node |
| 12 | Tree | root | address | Merupakan pointer yangmneunjuk ke root dari tree |
| 13 | PzlPosition | x | Point | Menjadi penampung data koordinat dari posisi elemen gambar puzzle |
| y | Point | Menjadi penampung data koordinat dari posisi elemen gambar puzzle |

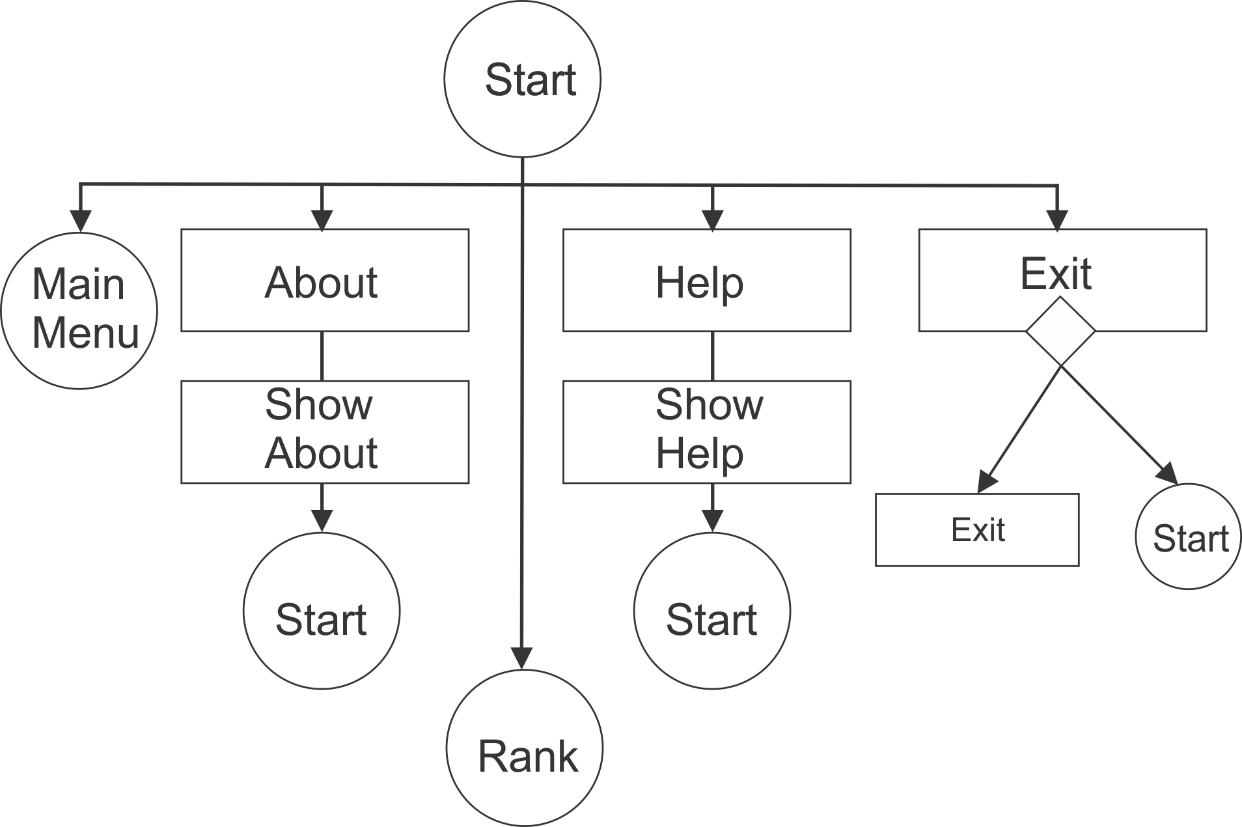
* **Modul**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Nama Modul** | **Deskripsi** | **Jenis** | **Parameter** | **Kamus Data (Lokal)** |
| 1 | Main Menu  **Pembuat :**   * Velia Sagita Putri (tampilan, pembuatan button) * Mufida Nuha Salimah (pewarnaan button) * Kiki Pratiwi (modul klik button) | Modul untuk menampilkan home | Prosedur |  | * MMsc, MMsc1, MMa, MMa1, MMh, MMh1, MMsg, MMsg1, MMe, MMe1 = bertipe point untuk menetukan titik button * mX, mY = bertipe integer untuk menentukan titik klik * valid : bertipe boolean dengan nilai awal false, untuk menghentikan looping klik mouse |
| 2 | About  **Pembuat :**   * Velia Sagita Putri (tampilan, pembuatan button) * Mufida Nuha Salimah (pewarnaan button) * Kiki Pratiwi (modul klik button) | Modul untuk membuat button ’about’ | Prosedur |  | * Bck1, Bck2 = bertipe point untuk menetukan titik button * mX, mY = bertipe integer untuk menentukan titik klik |
| 3 | Help  **Pembuat :**  Velia Sagita Putri | Modul untuk membuat button ’help’ | Prosedur |  | * Bck1, Bck2 = point untuk menetukan titik button * mX, mY = bertipe integer untuk menentukan titik klik |
| 4 | Set\_Back\_Button  **Pembuat :**   * Velia Sagita Putri (tampilan, pembuatan button) * Mufida Nuha Salimah (pewarnaan button) | Modul untuk membuat button ’back’ | Prosedur |  | * Bck1, Bck2 = bertipe point untuk menetukan titik button |
| 5 | Back\_Button\_Effect  **Pembuat :**   * Velia Sagita Putri (tampilan, pembuatan button) * Mufida Nuha Salimah (pewarnaan button) | Modul untuk memberi efek ditekan pada button ’back’ | Prosedur |  | * Bck1, Bck2 = bertipe point untuk menetukan titik button |
| 6 | Set\_Next\_Button  **Pembuat :**   * Velia Sagita Putri (tampilan, pembuatan button) * Mufida Nuha Salimah (pewarnaan button) | Modul untuk mampilkan button next yang bertujuan untuk kembali ke UI selanjutnya | Prosedur |  | * Bck1, Bck2 = bertipe point untuk menetukan titik button |
| 7 | Next\_Button\_Effect  **Pembuat :**   * Velia Sagita Putri (tampilan, pembuatan button) * Mufida Nuha Salimah (pewarnaan button) | Modul yang berisi perintah saat button next di-klik | Prosedur |  | * Bck1, Bck2 = bertipe point untuk menetukan titik button |
| 8 | Inputname  **Pembuat :**  Ananda Khoirunisa | Modul untuk menyimpan nama user | Prosedur | p = variabel input bertipe record player untuk menampung nama user |  |
| 9 | Result  **Pembuat :**  Ananda Khoirunisa | Modul yang ditampilkan setelah selesai memainkan game. Menampilkan skor pemain, pilihan menu high score, play game dan menu. | Prosedur |  | * Pg1, pg2, hg1, hg2, m1, m2 = bertipe point untuk menentukan titik button * mX, mY = bertipe integer untuk menentukan titik klik |
| 10 | Sound  **Pembuat :**  Ananda Khoirunisa | Modul untuk memainkan lagu di UI result | Prosedur |  |  |
| 11 | Find\_index  **Pembuat :**  Kiki Pratiwi | Modul untuk mendapatan indeks array | Fungsi | a[] = parameter input bertipe array of integer yang dicek apakah sama dengan nilai atau tidak  value = parameter input bertipe integer, diisi nilai array | * i = bertipe integer sebagai counter elemen array |
| 12 | Swap\_Position  **Pembuat :**   * Velia Sagita Putri (algoritma) * Kiki Pratiwi (revisi modul dengan pointer) | Modul untuk menukar isi array dengan array yang dituju saat menggerakkan kotak kosong | Fungsi | emptyArr = parameter bertipe pointer integer, diisi elemen array kosong  curposArr = parameter bertipe pointer integer, diisi oleh elemen array yang berisi nilai  no\_pic = parameter bertipe integer, diisi no gambar | * temp = bertipe integer, diisi sebagai wadah sementara saat pertukaran antara curposArr dengan emptyArr |
| 13 | Is\_Same\_As\_Goal  **Pembuat :**  Kiki Pratiwi | Modul untuk pengecekan apakah elemen telah sama dengan goal atau belum | Fungsi | arrPuzzle[] = parameter input bertipe array of integer, diisi array yang telah diisi berdasarkan move yang dilakukan dari user  arrGoal[] = parameter input bertipe array of integer, diisi array yang telah diinisiasi sebelumnya | * i = bertipe integer yang |
| 14 | random  **Pembuat :**  Kiki Pratiwi | Modul untuk assign angka ke array | Prosedur | Arr = parameter input/output tipe data array of integer | i = bertipe integer sebagai counter array  n = bertipe integer dengan niai awal 0 sebagai elemen array |
| 15 | Get\_Arr\_Puzzle\_Num  **Pembuat :**  Kiki Pratiwi | Modul untuk mengisi nomor puzzel dengan indeks array | Prosedur | Arr = parameter input/output tipe data array of integer  Pzl\_num[]= parameter input/output bertipe array of integer |  |
| 16 | Unique\_Random  **Pembuat :**  ”paxdiablo”  **Dimodifikasi oleh :**  Kiki Pratiwi | Modul untuk mengacak susunan pada puzzle | function | size = parameter input untuk menampung | i = bertipe integer sebagai counter array |
| 17 | letak  **Pembuat :**  Kiki Pratiwi | Modul untuk menentukan/ mendefinisikan titik acuan untuk membuat box pada papan puzzle. | Prosedur | Petak\_Puzzle = parameter input/output bertipe data Wadah untuk | i = bertipe data integer sebagai counter array  min\_x, min\_y = bertipe data integer untuk menampung nilai untuk membuat papan puzzle |
| 18 | Board  **Pembuat :**   * Kiki Pratiwi * Velia Sagita Putri (membuat patokan koordinat board game) | Modul untuk menempatkan gambar pada titik acuan yang telah ditentukan sebelumnya | Prosedur | t1 = Parameter input bertipe data Point , untuk menampung koodinat titik.  t2 = Parameter input bertipe data Point, untuk menampung koodinat titik.  pn = parameter input bertipe integer untuk | Str, pic\_num = bertipe data array of char |
| 19 | Goal  **Pembuat :**   * Velia Sagita Putri * Kiki Pratiwi (penambahan parameter) | Modul untuk menampilkan goal yang dipilih user sebagai panduan menyelesaikan puzzle | Prosedur | g = parameter input bertipe integer sebagai penentu gambar goal yang akan ditampilkan |  |
| 20 | ShowLevel  **Pembuat :**   * Velia Sagita Putri * Kiki Pratiwi (penambahan parameter) | Modul untuk menampilkan info level yang sebelumnya telah dipilih | Prosedur | lvl = parameter input bertipe pointer of character |  |
| 21 | Time\_icon  **Pembuat :**  Velia Sagita Putri | Modul untuk menampilkan image time | Prosedur |  |  |
| 22 | Showtime  **Pembuat :**   * Velia Sagita Putri(modul coundown time) * Kiki Pratiwi (modul menampilkan dalam bentuk grafis) | Modul untuk menampilkan countdown time dari waktu yang telah ditentukan yang sebelumnya telah dipilih | Prosedur | time\_left = parameter input bertipe record |  |
| 23 | Button\_Arrow  **Pembuat :**   * Velia Sagita Putri (pembuatan button) * Kiki Pratiwi (pembuatan parameter, modul klik) * Mufida Nuha Salimah (pewarnaan button) | Modul untuk membuat button arrow pada UI play | Prosedur | up1, up2, right1, right2, left1, left2, down1, down2 = parameter input/output bertipe pointer of integer untuk menampung titik acuan membuat button arrow |  |
| 24 | Button\_Menu\_Collection  **Pembuat :**   * Velia Sagita Putri (pembuatan button) * Kiki Pratiwi (pembuatan parameter, modul klik) * Mufida Nuha Salimah (pewarnaan button) | Modul untuk membuat button menu pada UI play | Prosedur | Pr, Pr1, Ps, Ps1, Pp, Pp1, Pb, Pb1, = parameter input/output bertipe pointer of record untuk menampung titik acuan membuat button menu |  |
| 25 | Place\_All\_Puzzle\_On\_board  **Pembuat :**  Kiki Pratiwi | Modul untuk memanggil gambar sesuai posisi setelah random | Prosedur | pp[] = parameter input bertipe record untuk mengirim posisi puzzle  ep[] = parameter input bertipe integer untuk mengirim posisi elemen |  |
| 26 | getInvCount  **Pembuat :**  Kiki Pratiwi | Modul untuk menghitung perpindahan elemn puzzle yang dilakukan user | Fungsi | arr[] = parameter input bertipe aray of integer | inv\_count = bertipe integer sebagai wadah counter perpindahan elemen  j = bertipe integer sebagai counter |
| 27 | isSolveable  **Pembuat :**  Kiki Pratiwi | Modul untuk menentukan apakah puzzle bisa diselesaikan | Fungsi | Puzzle[nPzl] = parameter input bertipe array of integer |  |
| 28 | solve\_puzzle  **Pembuat :**  Kiki Pratiwi | Modul untuk menyalin susunan puzzle di board menjadi susunan goal | Prosedur | Arr[nPzl] = parameter bertipe array of integer yang berisi susunan random puzzle  ArrGoal[nPzl] = parameter bertipe array of integer yang berisi susunan goal puzzle |  |
| 29 | set\_arr\_goal  **Pembuat :**  Kiki Pratiwi | Modul untuk menentukan goal puzzle | Prosedur | ArrGoal[nPzl] = parameter bertipe array of integer yang berisi susunan goal puzzle  ArrGoalSeting[nPzl] = parameter bertipe array of integer yang berisi salinan susunan goal puzzle |  |
| 30 | play\_game  **Pembuat :**   * Kiki Pratiwi * Velia Sagita Putri (coding tampilan, integrasi coundown time) * Mufida Nuha Salimah (pewarnaan button) | Modul untuk menjalankan proses gameplay. | Prosedur |  | up1, up2, right1, right2, left1, left2, down1, down2 = bertipe point untuk menampung titik acuan button arrow  Pr, Pr1, Ps, Ps1, Pp, Pp1, Pb, Pb1, = bertipe point untuk menampung titik acuan button menu  ArrGoal1, element\_puzzle, pzl\_num = bertipe data integer of array dengan maksimal 10 elemen  gm\_stg = bertipe record berisi setting yang telah dipilih user sebelumnya  plyr\_info = bertipe record berisi info user  win,valid,interupted, use\_help = bertipe boolean dengan nilai awal .False.  mX, mY = bertipe integer untuk menentukan posisi mouse click  no\_pic, cur\_pos = bertipe integer untuk menentukan gambar dan posisi elemen |
| 31 | ColorButton  **Pembuat :**  Mufida Nuha Salimah | Modul untuk mewarnai area dalam button atau box | Prosedur | t1 = Parameter input bertipe data Point , untuk menampung koodinat titik acuan pertama (kiri,atas)  t2 = Parameter input bertipe data Point, untuk menampung koodinat titik acuan kedua (kanan,bawah)  fillcolor1 = parameter input bertipe data integer, untuk menampung warna yang akan mengisi area bingkai button bagian atas dan kiri  fillcolor2 = Parameter input bertipe data integer, untuk menampung warna yang akan mengisi area bingkai button bagian bawah dan kanan  bordercolor1 = parameter input bertipe data integer, untuk menampung warna yang akan menjadi parameter input dalam modul floodfill(modul dalam graphics.h) yang berfungsi sebagai acuan warna akan mengisi sampai mana. | Variabel x1, y1 ,x2 ,y2 bertipe integer untuk menampung nilai koordinat |
| 32 | Rank  **Pembuat :**  Mufida Nuha Salimah | Modul yang menampilkan rank pemain/ leaderboard berdasarkan 3 level | Prosedur |  | * Re, Re1, Rm, Rm1, Rh, Rh1 = bertipe point dengan subvar x dan y untuk menetukan titik button * mX, mY = bertipe integer untuk menentukan titik acuan klik * valid = variabel bertipe boolean dengan nilai awal false, untuk menghentikan looping klik mouse |
| 33 | Box  **Pembuat :**  Velia Sagita Putri | Modul untuk membuat persegi | Prosedur | t1 = Parameter input bertipe data Point , untuk menampung koodinat titik.  t2 = Parameter input bertipe data Point, untuk menampung koodinat titik. |  |
| 34 | Button  **Pembuat :**  Velia Sagita Putri | Modul untuk membuat button | Prosedur | t1 = Parameter input bertipe data Point , untuk menampung koodinat titik.  t2 = Parameter input bertipe data Point, untuk menampung koodinat titik. | t3, t4 = bertipe record point dengan subvar x dan y untuk menentukan titik button bagian luar |
| 35 | Button\_Image  **Pembuat :**   * Velia Sagita Putri(menentukan titik koordinat) * Mufida Nuha Salimah (menambahkan parameter image, gap, dan gap\_image) | Modul untuk membuat button yang terdapat gambar | Prosedur | t1 = Parameter input bertipe data Point , untuk menampung koodinat titik.  t2 = Parameter input bertipe data Point, untuk menampung koodinat titik.  image = Parameter input/output bertipe pointer of constanta character yang menampung alamat file pada folder.  gap, gap\_image = parameter bertipe integer yang merepresentasikan jarak antara persegi dalam dan luar pada button (gap), dan jarak button dengan gambar (gap\_image) | t3, t4 = variabel bentukan bertipe record point dengan subvar x dan y untuk menentukan titik button. |
| 36 | splash\_screen  **Pembuat :**  Kiki Pratiwi | Modul untuk menampilkan screen awal aplikasi | Prosedur |  | width = bertipe integer dengan nilai awal 0  str = bertipe array of character dengan 24 element sebagai penampung nama file  pic\_num = bertipe array of character dengan 3 elemen character sebagai penampung urutan huruf |
| 37 | save\_data\_player  **Pembuat :**  Kiki Pratiwi | Modul untuk menyimpan data ke file | Prosedur | p = parameter input bertipe record player yang berisi info user | fplayer = pointer bertipe file  chars = pointer bertipe integer sebagai testcase jumlah data yang tersimpan |
| 38 | get\_data\_lates \_player  **Pembuat :**  Kiki Pratiwi | Modul untuk menampilkan file pemain terakhir | Prosedur | p = parameter input bertipe pointer record player yang berisi info user | fplayer = pointer bertipe file  file\_p = bertipe record player sebagai penampung data |
| 39 | pause\_act  **Pembuat :**  Kiki Pratiwi | Modul untuk menghentikan coundown time | Prosedur | Pp, Pp1 = parameter input bertipe record point sebagai penentu koordinat klik | mX,mY = bertipe integer untuk menentukan area yang bisa di klik  valid = bertipe boolean dengan ilai awal .False. sebagai penghenti looping |
| 40 | set\_goal\_and\_level  **Pembuat :**  Kiki Pratiwi | Modul untuk menentukan goal dan level | Prosedur |  | mX,mY = bertipe integer untuk menentukan area yang bisa di klik  valid, next\_clicked = bertipe boolean dengan ilai awal .False. sebagai penghenti looping  E1, E2, M1, M2, H1, H2, ga1, ga2, gb1, gb2, gc1,gc2,Bck1, Bck2, nxt1, nxt2 = bertipe record point sebagai penentu koordinat button  ArrGoal1[nPzl], ArrGoal2[nPzl], ArrGoal3[nPzl]  = konstanta bertipe array of integer sebagai default goal  width, width\_ = bertipe integer untuk menentukan jarak antara pilihan level dan goal |
| 41 | save\_setting  **Pembuat :**  Nikita Nabila | Modul untuk menyimpan goal level yang dipilih pemain ke file | Prosedur | gs = parameter bertipe record game\_setting | goal\_and\_level\_setting = bertipe pointer of file untuk menyimpan data level dan goal yang dipilih user |
| 42 | get­\_game\_setting  **Pembuat :**  Nikita Nabila | Modul untuk memanggil file goal dan level yang dipiliih user dari file | Prosedur | gs = parameter bertipe record game\_setting | file\_gs = bertipe record game setting untuk wadah data dari file  goal\_and\_level\_setting = bertipe pointer of file untuk menyimpan data level dan goal yang dipilih user |
| 43 | praexit  **Pembuat :**  Nikita Nabila | Modul untuk menanyakap apakan user akan benar-benar keluar dari games | Prosedur |  | EX1, EX2, YEX1, YEX1, NEX1, NEX1 **=** bertipe record point untuk penentu koordinat button  mX, mY = bertipe integer untuk menentukan koordinat yang bisa diklik |
| 44 | Input\_Custom  **Pembuat :**  Velia Sagita Putri | Modul untuk menyimpan data custom goal | Fungsi | Player\_setting = parameter bertipe game\_setting |  |
| 45 | BackTrack  **Pembuat :**  Velia Sagita Putri | Modul untuk backtrack dari step terakhir menuju goal pada tree dengan metode stack | Fungsi | Root = parameter bertipe pointer yang merupakan dari root tree  Leaf = parameter bertipe addressLink yang merupakanposisi node yang berisi posisi step terakhir pemain | Stack = bertipe addressLink yang merupakan stack  newNode = bertipe addressLink yang merupakan node yang nantinya akan disambungkan ke node yang ada  Pcur = bertipe pointer yang menunjuk info dari leaf  Node = bertipe address yang merupakan pointer yang menunjuk pcur |
| 46 | randomLeaf  **Pembuat :**  Velia Sagita Putri | Modul untuk random leaf dari level yang dipilih | Fungsi | Front = parameter bertipe pointer yang merupakan leaf terkiri dari level tersebut  Leaf = parameter bertipe integer | LeafNumber = bertipe integer untuk penampung urutan node keberapa yang ditampilkan pada hasil random  i = bertipe integer untuk counter pengulangan perpindahan pointer  pCur = bertipe addressLink berupa pointer yang merupakan penunjuk yang pindah menuju hasil random |
| 47. | Show\_Rank  **Pembuat :**  Mufida Nuha Salimah | Modul untuk menampilkan urutan *user* terbaik berdasarkan waktu untuk menyelesaikan *game*. | Prosedur | level = parameter bertipe string yang merepresentasikan level yang telah dipilih oleh *user*  colorfont = parameter bertipe integer yang merepresentasikan warna huruf yang ditampilkan | FILE \*fplayer,,  player temp,,  int count, xnama, yscore, xscore, xlevel, skyBlue,,  char printf[30], str\_name[7], str\_movement[3], str\_timeleft[6], |
| 48. | set\_custom\_goal  **Pembuat :**  Mufida Nuha Salimah | Modul untuk membuat *goal* *custom* | Prosedur | - | PzlPosition coordinate[24],,  char path[28],,  int alphabet\_initial,  orange1, orange2, lightGreen1, lightGreen2, width, height, gap\_x, gap\_button  Point resetButton1, resetButton2, doneButton1, doneButton2, backButton1,  backButton2,, |
| 49. | back\_action  **Pembuat :**  Kiki Pratiwi | Modul untuk kembali ke UI sebelumnya | Prosedur | int mX, mY | Point backButton1,  backButton2,, int width, height, gap\_x, gap\_button, gapImage |
| 50. | reset\_action  **Pembuat :**  Kiki Pratiwi | Modul untuk *reset* gambar pada *custom goal* | Prosedur | int mX, mY,, char arr\_custom\_setting[] | PzlPosition Petak\_Puzzle[nPzl+1],, idx\_pzl[nPzl |
| 51. | Pic\_Position  **Pembuat :**  Mufida Nuha Salimah | Modul untuk mendefinisikan koordinate gambar | Prosedur | PzlPosition coor | x dan y bertipe integer sebagai counter pertambahan nilai |
| 52. | choose\_pic  **Pembuat :**  Mufida Nuha Salimah | Modul untuk memilih gambar pada *set\_custom\_goal* | Fungsi | Char ArrGoal[] | coordinate[24] bertipe data komposit PzlPosition yang berisi informasi posisi puzzle,  path[28], cek[24], img[24] bertipe karakter sebagai penampung letak gambar,  orange1, orange2, lightGreen1, width, lightGreen2 bertipe integer untuk menampung informasi jenis warna height, gap\_x, gap\_button, gapImage bertipe integer untuk menampung informasi jarak antar button  resetButton1, resetButton2, doneButton1, doneButton2, backButton1,  backButton2 bertipe data komposit Point sebagai acuan titik suatu button |
| 53. | Box\_Pic  **Pembuat :**  Mufida Nuha Salimah | Modul untuk mewarnai border diantara dua persegi | Prosedur | Color = parameter input bertipe integer yang merepresentasikan warna yang akan digunakan.  dot1, dot2 = parameter input bertipe data komposit yang yang berisi info titik yang menjadi batas pewarnaan |  |
| 54. | movementFreeSpace  **Pembuat :**  Mufida Nuha Salimah | Modul yang berfungsi mengembalikan tipe data komposit yaitu ’*infoMovement*’ yang memberi info perpindahan yang dapat dilakukan ’*FreeSpace*’ | Fungsi | indexPuzzle = parameter input bertipe integer yang berisi info indeks yang diisi info pergerakannya pada papan puzzle | freeSpace bertipe data komposit infoMovement yang berisi info pergerakan yang dapat dilakukan. |
| 55. | set\_info\_tree\_nodeRoot  **Pembuat :**  Mufida Nuha Salimah | Mengembalikan tipe data komposit ’*infotype’* yang berisi *move, reverse,* dan indeks *empty* Root. | Fungsi | indexEmpty = parameter input bertipe integer yang berisi info indeks yang kosong pada papan puzzle | root = bertipe data infotype yang menampung info dari node root pada tree |
| 56. | createRoot  **Pembuat :**  Mufida Nuha Salimah | Mengembalikan tipe data komposit ’*address´* atau *’infotypeLink’* yang menjadi root pada Tree yang dibangun | Fungsi | indexEmpty = parameter input bertipe integer yang berisi info indeks yang kosong pada papan puzzle | root = bertipe data address sebagai penampung node baru yang akan dibuat |
| 57. | generate\_tree  **Pembuat :**  Kiki Pratiwi | Membangun Tree yang mengembalikan tipe data komposit ’*tree\_and\_leaf*’ yang berisi Tree yang telah dibangun dan Queue yang berisi *leaf*. | Fungsi | ArrGoal parameter input array satu dimensi bertipe data karakter yang berisi info array goal  depth |  |
| 58. | set\_info\_tree\_node  **Pembuat :**  Kiki Pratiwi | Mengembalikan tipe data komposit ’*infotype’* yang berisi *move, reverse,* , dan indeks *empty* node baru. | Fungsi | Short int arah, empty |  |
| 59 | animatePuzzle  **Pembuat :**  Mufida Nuha Salimah | Menggerakkan puzzle secara otomatis berdasarkan info pergerakan dari stack | Prosedur | addressLink topStack,, char ArrGoal[], int ArrIdx[] | Petak\_Puzzle[nPzl] array satu dimensi berjumlah nPzl bertipe data PzlPosition yang menampung info posisi papan puzzle idx\_empty, idx\_curpos bertipe data integer sebagai penampung indeks papan puzzle yang kosong  info bertipe data komposit infotypeLink, |
| 60. | deAlokasiStack  **Pembuat :** Mufida Nuha Salimah | Melakukan dealokasi tiap-tiap node pada seluruh stack | Fungsi | addressLink pDel |  |
| 61. | isEmpty  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengecek isi suatu node, mengembalikkan nilai boolean | Fungsi | addressLink p |  |
| 62. | alokasiNode  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengalikasikan sejumlah memori yang dipesan, mengembalikan nilai addressLink | Fungsi |  | addressLink newNode |
| 63. | createNode  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Membuat node baru, mengembalikan nilai addressLink | Fungsi | infotypeLink info | addressLink newNode |
| 64. | insertAwal  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Menambah / insert node baru pada di awal Linked List | Prosedur | addressLink awal, newNode |  |
| 65. | insertAkhir  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Menambah / insert node baru pada di akhir Linked List | Prosedur | addressLink awal, akhir, newNode |  |
| 66. | Enque  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Menambah / insert node baru pada di akhir Linked List pada stuktur data Queue | Prosedur | addressLink awal, akhir  infotypeLink info | addressLink pNew |
| 67. | Push  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Menambah / insert node baru pada di awal Linked List pada stuktur data Stack | Prosedur | addressLink awal, newNode |  |
| 68. | deAlokasi  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Melakukan dealokasi dan pengembalian address pDel ke sistem | Prosedur | addressLink pDel |  |
| 69. | deleteAwal  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengehapus elemen pertama pada Linked List dan mengembalikan nilai ’*infotypeLink’* | Prosedur | addressLink node | infotypeLink deleteNode, addressLink pDel |
| 70. | Deque  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengehapus elemen pertama pada Linked List dan mengembalikan nilai ’*infotypeLink’* pada struktur data Queue | Fungsi | addressLink nodeFront | infotypeLink deleteNode |
| 71. | Pop  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengehapus elemen pertama pada Linked List dan mengembalikan nilai ’*infotypeLink’* pada struktur data Stack | Fungsi | addressLink top | infotypeLink deleteNode |
| 72. | IsEmpty  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengecek isi suatu node, mengembalikkan nilai boolean | Fungsi | Address p |  |
| 73. | alokasiNodeTree  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Mengalikasikan sejumlah memori yang dipesan, mengembalikkan nilai address | Fungsi |  | Address newNode |
| 74. | createNodeTree  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Membuat node baru, mengembalikan nilai address | Fungsi | Infotype info | Address newNode |
| 75. | insertNodeTree  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Menambah / insert node baru pada Tree | Fungsi | Tree tree,, address parent,, infotype info | Address newNode, temp |
|  | createTree  **Pembuat :**  AN  **Dimodifikasi oleh :**  Mufida Nuha Salimah | Membuat (Root) Tree baru | Prosedur | Tree tree |  |
| 76. | deAlokasiStack  **Pembuat :**  Mufida Nuha Salimah | Mendealokasi stack | Fungsi | addressLink pDel | infotypeLink deleteNode |
| 77. | cheat\_input\_custom  **Pembuat :**  Kiki Pratiwi | Mengisi array puzzle custom secara otomatis | Fungsi | Int empty | PzlPosition Petak\_Puzzle[nPzl], game\_setting custom\_setting, char ArrGoal3[nPzl], int idx\_pzl[nPzl], PzlPosition coordinate[24] |
| 78. | show\_all\_button\_set\_goal\_and\_level  **Pembuat :**  Kiki Pratiwi | Menampilkan semua button pada screen set goal and level | Prosedur | char\* level, char goal | Point E1, E2, M1, M2, H1, H2, ga1, ga2, gb1, gb2, gc1, gc2, Bck1, Bck2, nxt1, nxt2. |
| 79. | count\_data\_player  **Pembuat :**  Kiki Pratiwi | Menghitung data player yang tersimpan File | Fungsi |  | player data, FILE \*fplayer, int tot\_data |
| 80. | find\_index\_of\_empty\_puzzle  **Pembuat :**  Kiki Pratiwi | Menemukan index array yang berisi EMPTY\_PUZZLE | Fungsi | char a[] | Int i |
| 81. | get\_random\_number  **Pembuat :**  Kiki Pratiwi | Mendapatkan angka acak pada batas tertentu | Fungsi | int limit\_a, int limit\_b) | unsigned short N |
| 82. | Loading  **Pembuat :**  Ananda Khoirunisa | Menampilkan screen loading saat generate tree | Prosedur |  |  |
| 83. | randomGoal  **Pembuat :**  Ananda Khoirunisa | Memilih ulang node yang ditampilkan, lalu menembalikkan node yang telah dipilih dengan tipe data addressLink | Fungsi | addressLink stack, char ArrGoal[], int Arridx\_ofGoal[] | Int idx\_empty, idx\_temp, idx\_reverse  addressLink nodeCurrent, stackBacktrack, pDel, newNode  infotypeLink info  PzlPosition Petak\_Puzzle[nPzl] |
| 84. | Input\_empty\_puzzle  **Pembuat :**  Ananda Khoirunisa | Menambahkan array untuk diisi pada custum goal | Fungsi | PzlPosition pp[nPzl] | Point Back1, Back2  Int width, heigth, gap\_x, gap\_button, gapImage, i, idx\_empty, mX, mY  Boolean valid, valid\_area |
| 85. | isArrFull  **Pembuat :**  Ananda Khoirunisa | Mengecek isi array goal telah penuh atau belum yang mengembalikkan tipe data boolean | Fungsi | Char array[] | Boolean full  Unsigned short nfilled |
| 86. | Reset\_arr\_custom  **Pembuat :**  Ananda Khoirunisa | Mengembalikan nilai array goal menjadi kosong kembali | Prosedur | Char array[] | Int i |
| 87. | Get\_leaf  **Pembuat :**  Nikita Nabila | Mengambil leaf yang telah ditentukan nilai randomnya, lalu mengembalikkan address | Fungsi | Address nodeRoot | Address pCur  Boolean arah  addressLink Front, Rear |
| 88. | Show\_Image\_Goal\_Custom  **Pembuat :**  Nikita Nabila | Menampilkan goal pada layar permainan | Prosedur | Char ArrGoal | Point SX1, SX2  PzlPosition Petak\_Puzzle[nPzl] |
| 89. | Set\_Coordinate\_Image\_Goal\_Custom  **Pembuat :**  Nikita Nabila | Mendeklarasikan koordinat yang digunakan untuk menampilkan goal pada layar permainan | Prosedur | PzlPosition Petak\_Puzzle [nPzl] | Int min\_y, min\_x |
| 90. | Countleaf  **Pembuat :**  Nikita Nabila | Menghitung jumlah daun pada tree yang telah dibuat lalu mengambalikan jumlah leaf yang telah dihitung | Fungsi | addressLink node | Int countleaf |

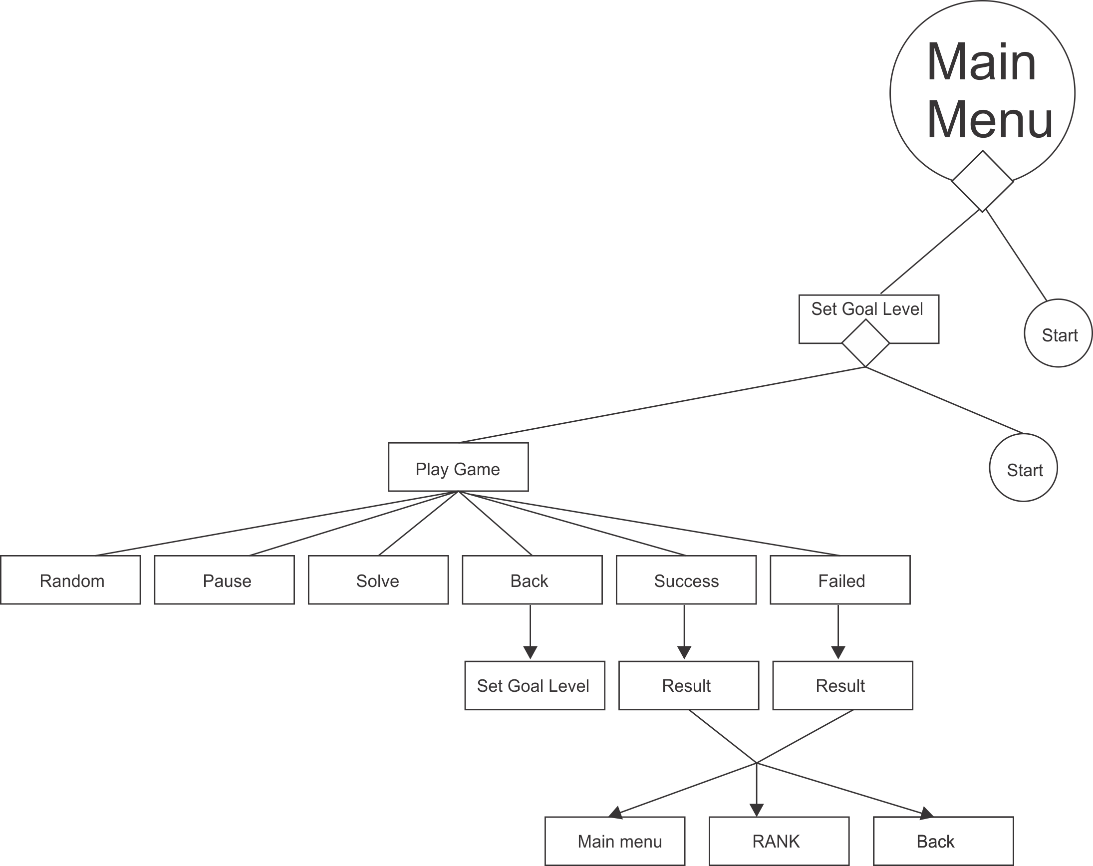
### Perancangan Struktur Menu

****

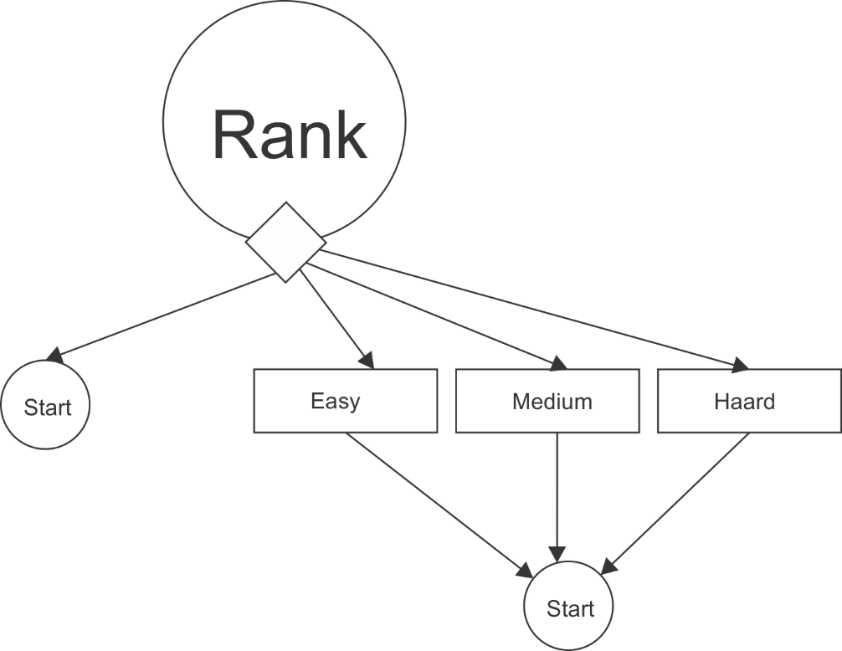
Gambar 1 struktur splashscreen

****

Gambar 2 struktur main menu

****

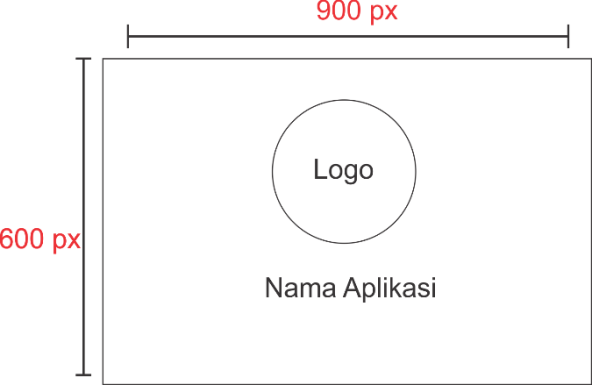
Gambar 3 struktur set goal and level & play

****

Gambar 4 struktur rank

### Desain GUI

1. **Splash Screen**

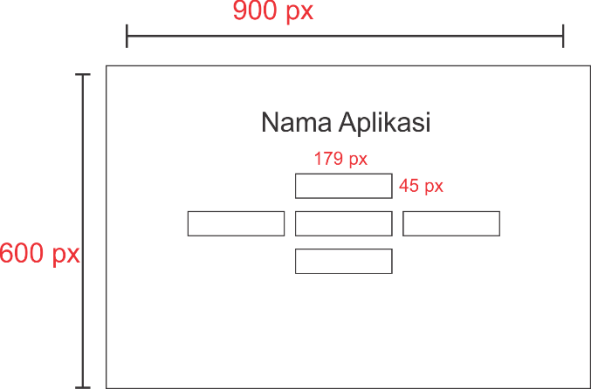
Gambar 5 GUI splash screen

Design : Kiki Pratiwi & Ananada Khoirunisa

Tool Design : CorelDraw X7

Coding : Kiki Pratiwi

1. **Main Menu**

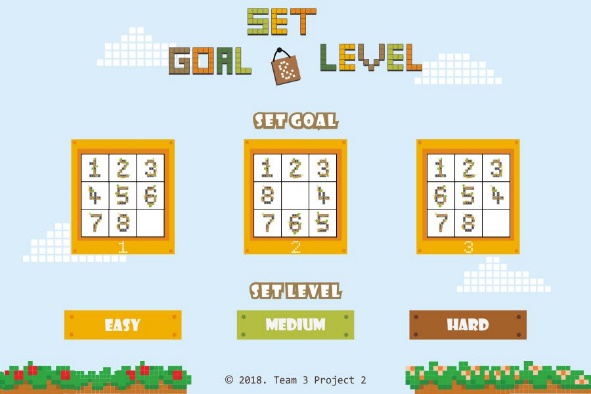
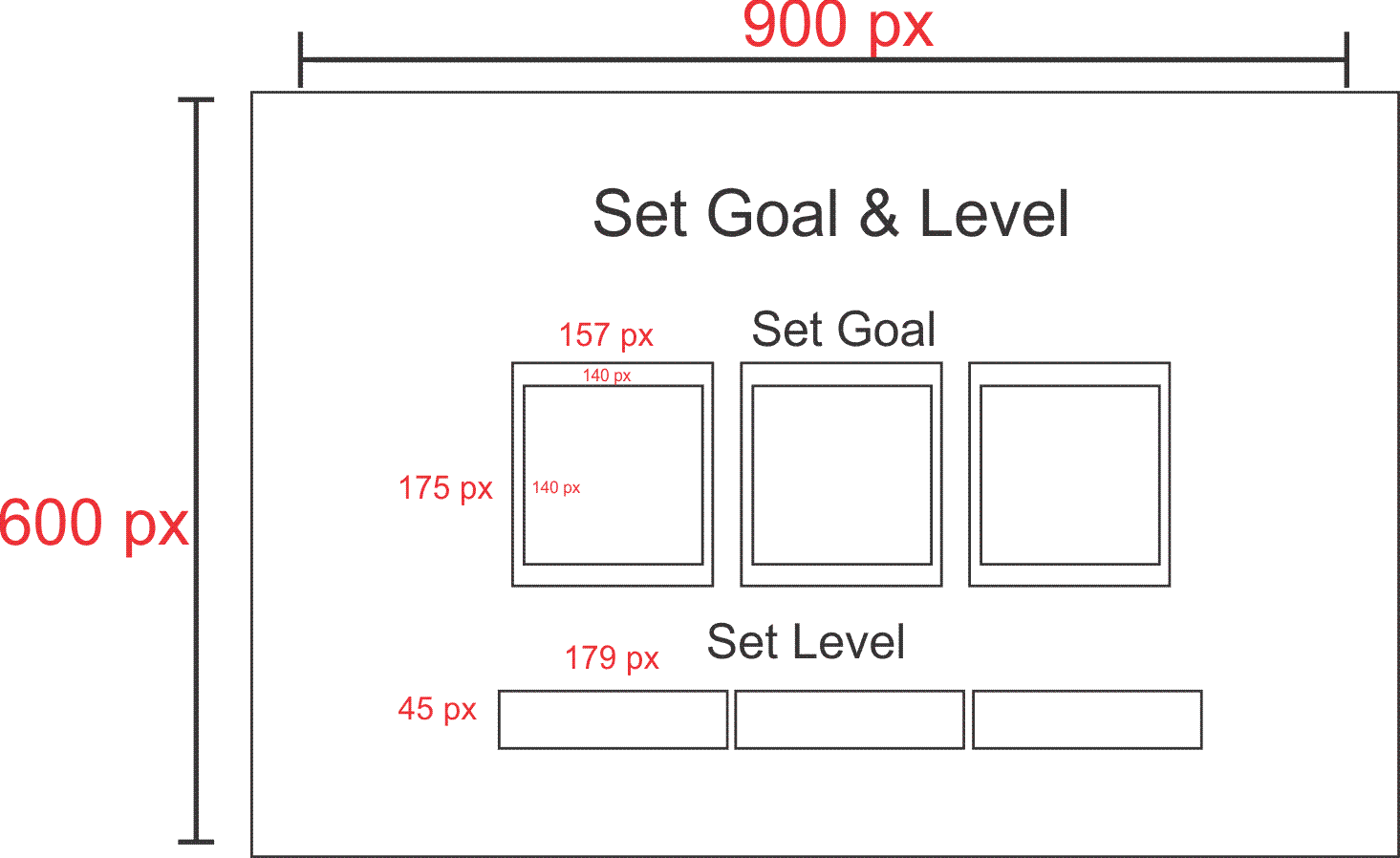
Gambar 6 GUI Main Menu

Design : Kiki Pratiwi dan Ananda Khoirunisa

Tool Design : CorelDraw X7

Coding : Velia Sagita Putri

1. **Set Goal and Level**

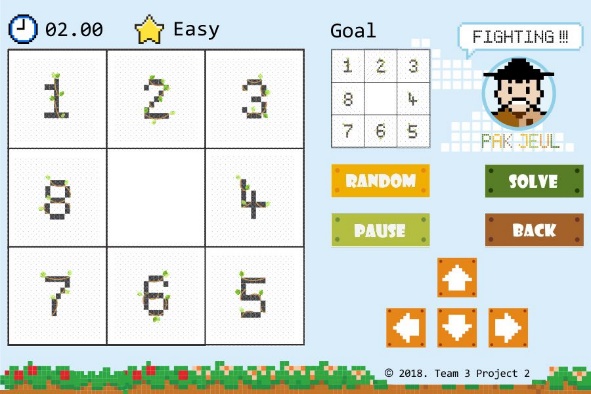
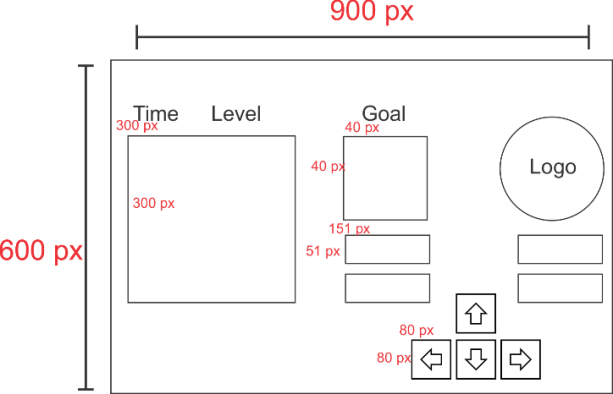
Gambar 7 GUI set goal and level

Design : Kiki Pratiwi & Ananda Khoirunisa

Tool Design : CorelDraw X7

Coding : Kiki Pratiwi

1. **Playgame Screen**

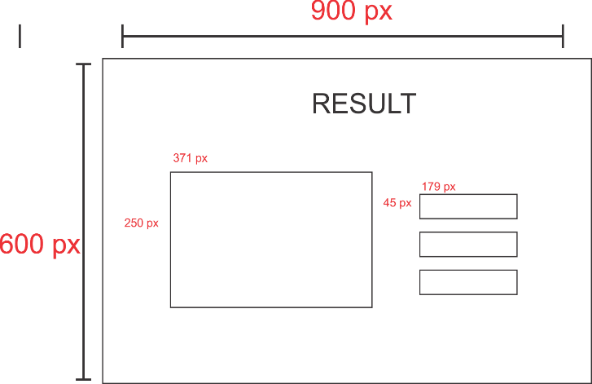
Gambar 8 GUI play game

Design : Kiki Pratiwi & Ananda Khorunisa

Tool Design : CorelDraw X7

Coding : Velia Sagita Putri

1. **Result Screen**

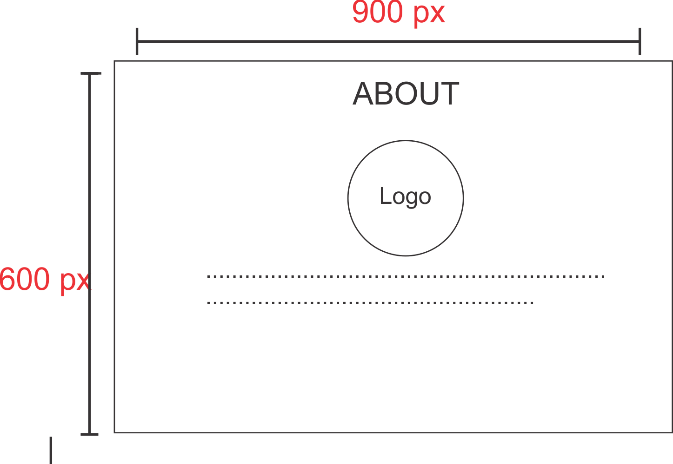
Gambar 9 GUI result screen

Design : Kiki Pratiwi & Ananda Khorunisa

Tool Design : CorelDraw X7

Coding : Ananda Khoirunisa

1. **About Screen**



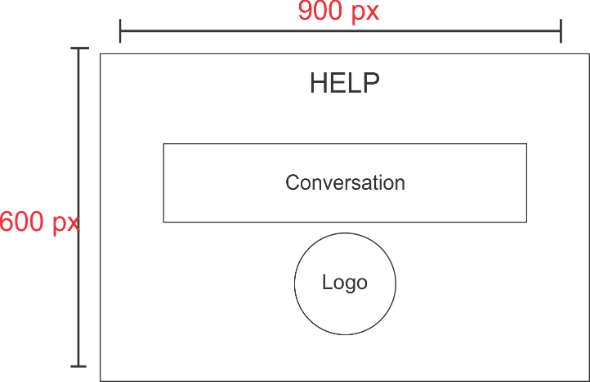
Gambar 10 GUI about screen

Design : Kiki Pratiwi & Ananda Khorunisa

Tool Design : CorelDraw X7

Coding : Velia Sagita Putri

1. **Help Screen**

Gambar 11 GUI help screen

Design : Kiki Pratiwi & Ananda Khorunisa

Tool Design : CorelDraw X7

Coding : Velia Sagita Putri

1. **Rank Screen**

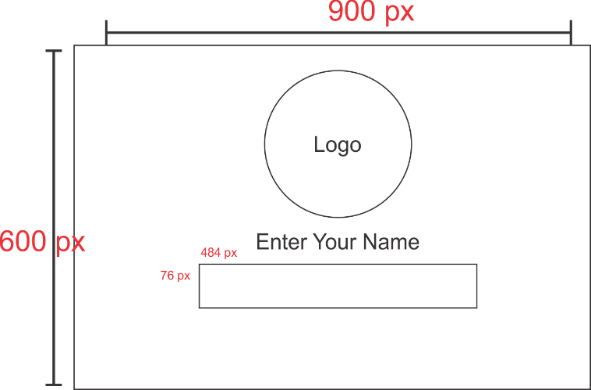
Gambar 12 GUI rank screen

Design : Kiki Pratiwi & Ananda Khorunisa

Tool Design : CorelDraw X7

Coding : Mufida Nuha Salimah

1. **Input Name Screen**



Gambar 13 GUI input name screen

Design : Kiki Pratiwi & Ananda Khorunisa

Tool Design : CorelDraw X7

Coding : Ananda Khoirunisa

1. **Tools yang digunakan**
   * + 1. DevC++
       2. Microsoft Word
       3. Microsoft Excel
       4. Microsoft Power Point
       5. CorelDrawX7
       6. Photoshop CC
       7. Github
       8. Trello
       9. Notepad
       10. LINE
       11. Google Drive
       12. Google Spreadsheet

# **BAB III**

**EVALUASI**

## **Lesson Learn**

* Graphics.h adalah header yang mengubah format text yang menjadi bawaan Bahasa C
* Install graphics.h harus ditambah library libbgi.a dan parameter additional lgbi, lgdi32, lcomdlg32, luuid, loleaut32, dan lole32
* floodfill(int x,int y, int border) berguna untuk menentukan titik acuan bagian objek yang akan diwarnai dan sampai bertemu suatu warna berhenti.
* readimagefile(\*namafile,left,top,right,bottom) berguna untuk menambahkan gambar ke program dengan mengatur kuruan lewat letak titik sudutnya.
* Dalam pembuatan titik dibutuhkan 2 koordinat, yaitu x dan y.
* Dalam pembuatan garis dibutuhkan 2 titik dengan koordinat (x1,y1) dan (x2,y2).
* Dalam pembuatan box dibutuhkan 4 titik dengan koordinat (x1,y1), (x2,y1), (x1,y2), dan (x2,y2).
* Dalam pembuatan button dibutuhkan :
* 8 titik dengan koordinat (x1,y1), (x2,y1), (x1,y2), (x2,y2), (x3,y3), (x4,y3), (x3,y4), dan (x4,y4)
* 2 box (box 1 dengan koordinat (x1,y1), (x2,y1), (x1,y2), dan (x2,y2) dan box 2 dengan bkoordinat(x3,y3), (x4,y3), (x3,y4), dan (x4,y4))
* 12 line dengan koordinat ((x1,y1), (x2,y1)), ((x1,y1), (x1,y2)), ((x1,y1), (x2,y2)),((x1,y1), (x2,y1)), ((x3,y3), (x4,y3)), ((x3,y3), (x3,y4)), ((x3,y3), (x4,y4)),((x3,y3), (x4,y3)), ((x1,y1), (x3,y3)), ((x2,y1), (x4,y3)), ((x1,y2), (x3,y4)),((x2,y2), (x4,y4))
* setcolor(warna) berguna untuk mengganti warna garis pada objek.
* setfillstyle(int patern, int color) berguna untuk mengganti jenis patern dalam objek dan warna patern yang akan digunakan.
* Setrgbpalette(int colournum, int red, int green, int blue) fungsi ini berguna untuk menyetting warna sesuai dengan kemauan.
* Playsound (TEXT, (.wav"),NULL,SND\_ASYNC | SND\_LOOP)) untuk menjalankan lagu dalam program dengan ekstensi wav
* Penambahan gambar hanya bisa dalam format bitmap saja.
* Dalam teleworking dibutuhkan koneksi internet yang stabil dan kerjasama juga kesabaran, terutama saat menunggu compile sekian lama namun ternyata masih bug
* Ukuran di CorelDraw dengan IDE DevC++ tidak sama
* Pada modular programing sangat ditekankan untuk memnggunakan variabel lokal agar tidak ada multiple declaration dan bug
* Dalam setiap modul bisa disederhanakan lagi dengan penggunaan parameter agar modul yang memiliki fungsi sama bisa dibuat 1 modul

# **BAB IV**

**PENUTUP**

## **Kesimpulan**

Berdasarkan target yang ditentukan yaitu membuat 8-puzzle dengan metode array berhasil terselesaikan bahkan ditambah dengan integrase grafik dan sound.

Spesifikasi yang telah dicapai :

1. Pemilihan level
2. Menampilkan countdown time
3. Random puzzle
4. Save data player ke file
5. Pause games
6. Backsound di tiap screen
7. Controller dengan mouse
8. Penerapan button
9. Pemilihan grafis yang menarik
10. Solve puzzle (hint)
11. **Saran**

Masih banyak kekurangan dari game 8-puzzle yang telah dibuat, maka akan dikembangankan, diantaranya :

1. Menggunakan keyboard sebagai *button controller* tambahan

**LAMPIRAN**

## **Progress Kerja**

### Pertemuan Ke-1 (15 Februari 2018)

* **Pelaksanaan :**
* Pembagian kelompok
* Penentuan konsep e-logbook
* Eksplorasi install header graphics.h
* **Pencapaian :**
* Berhasil menginstall header graphics.h
* Berhasil membuat circle dengan graphics.h
* Berhasil membuat line dengan graphics.h
* **Masalah :**
* Setelah IDE di close, installasi header terhapus
* Terjadi bug saat compile
* **Solusi :**
* Saat membuka lagi IDE, ada ask screen pilih ‘No’ agar setting yang telah diatur tidak hilang
* Menambahkan library dan parameter tambahan di compiler setting
* **Outcome :**
* Graphics.h adalah header yang mengubah format text yang menjadi bawaan Bahasa C
* Install graphics.h harus ditambah library libbgi.a dan parameter additional lgbi, lgdi32, lcomdlg32, luuid, loleaut32, dan lole32
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Install graphics.h
* Membuat objek dengan header graphics.h
* Kiki Pratiwi
* Install graphics.h
* Membuat objek dengan header graphics.h
* Mufida Nuha Salimah
* Install graphics.h
* Membuat objek dengan header graphics.h
* Nikita Nabila
* Install graphics.h
* Membuat objek dengan header graphics.h
* Velia Sagita Putri
* Install graphics.h
* Membuat objek dengan header graphics.h

### Pertemuan Ke-2 (22 Februari 2018)

* **Pelaksanaan :**
* Membuat struck titik
* Membuat box dengan struck titik dan fungsi line dari graphics.h
* Membuat button dengan struct titik, fungsi line dan modul box yang telah dibuat
* Membuat text dengan fungsi outtextxy
* Memberi warna line dan objek
* Membuat button dengan efek 3D
* Menambahkan gambar dengan modul getimagefile
* **Pencapaian :**
* Berhasil membuat struct titik
* Berhasil menbuat box dengan struct titik
* Berhasil membuat button dengan struct titik, fungsi line dan modul box yang telah dibuat
* Berhasil membuat text dengan fungsi outtextxy
* Berhasil memberi warna line dan objek
* Berhasil membuat button dengan efek 3D
* **Masalah :**
* Warna terkadang meleber ke bagian yang tidak seharusnya diwarnai
* Button belum bisa ada efek klik karena sulit menentukan area yg clickable
* Sulit membuat garis antara sudut box dalam dan sudut box atas untuk menjadikan button 3D
* **Solusi :**
* Agar warna tidak meleber titik yang ditentukan adalah titik tengah objek dan batas akhir dengan menentukan warna garis yang bersentuhan
* Button bisa di klik dengan area titik sudut terluar dari button
* Pembuatan garis penyambung antara box dalam dan luar dengan membuat line dengan titik bagian sudut yang paling berdekatan, misalkan kanan atas box besar dengan kanan atas box kecil
* **Outcome :**
* floodfill(int x,int y, int border) berguna untuk menentukan titik acuan bagian objek yang akan diwarnai dan sampai bertemu suatu warna berhenti.
* readimagefile(\*namafile,left,top,right,bottom) berguna untuk menambahkan gambar ke program dengan mengatur kuruan lewat letak titik sudutnya.
* Dalam pembuatan titik dibutuhkan 2 koordinat, yaitu x dan y.
* Dalam pembuatan garis dibutuhkan 2 titik dengan koordinat (x1,y1) dan (x2,y2).
* Dalam pembuatan box dibutuhkan 4 titik dengan koordinat (x1,y1), (x2,y1), (x1,y2), dan (x2,y2).
* Dalam pembuatan button dibutuhkan :

1. 8 titik dengan koordinat (x1,y1), (x2,y1), (x1,y2), (x2,y2), (x3,y3), (x4,y3), (x3,y4), dan (x4,y4)
2. 2 box (box 1 dengan koordinat (x1,y1), (x2,y1), (x1,y2), dan (x2,y2) dan box 2 dengan bkoordinat(x3,y3), (x4,y3), (x3,y4), dan (x4,y4))
3. 12 line dengan koordinat ((x1,y1), (x2,y1)), ((x1,y1), (x1,y2)), ((x1,y1), (x2,y2)),((x1,y1), (x2,y1)), ((x3,y3), (x4,y3)), ((x3,y3), (x3,y4)), ((x3,y3), (x4,y4)),((x3,y3), (x4,y3)), ((x1,y1), (x3,y3)), ((x2,y1), (x4,y3)), ((x1,y2), (x3,y4)),((x2,y2), (x4,y4))

* setcolor(warna) berguna untuk mengganti warna garis pada objek.
* setfillstyle(int patern, int color) berguna untuk mengganti jenis patern dalam objek dan warna patern yang akan digunakan.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Explorasi coloring object
* Kiki Pratiwi
* Explorasi button
* Mufida Nuha Salimah
* Explorasi coloring object
* Nikita Nabila
* Explorasi titik koordinat siatu objek
* Velia Sagita Putri
* Explorasi button

### Pertemuan Ke-3 (1 Maret 2018)

* **Pelaksanaan :**
* Mendesign layout pada kertas
* Membuat layout pada IDE DevC++
* Menggunakan trello untuk logbook
* Mengaplikasikan tone warna RGB
* Mendesign elemen puzzle
* Menginput lagu ke program
* Menentukan karakter puzzle
* **Pencapaian :**
* Berhasil mendesain layout game pada kertas,
* Berhasil menggunakan trello untuk membuat e-logbook,
* Berhasil mengaplikasikan variasi warna RGB,
* Berhasil mendesain layout game pada program,
* Berhasil mendesain elemen-elemen pada puzzle game,
* Berhasil menginput lagu ke dalam program,
* Berhasil menentukan karakter dan alur dari game.
* **Masalah :**
* Kebingungan menentukan fungsi RGB
* Kebingungan menentukan alur game yang pas
* kebingungan menentukan karakter puzzle
* **Solusi :**
* Menggunakan bantuan aplikasi colorpic untuk menentukan besaran RGB
* Mengeksplore lagi games edge puzzle yang ada
* Menentukan karakter dengan konsep pertanian
* **Outcome :**
* Setrgbpalette(int colournum, int red, int green, int blue) fungsi ini berguna untuk menyetting warna sesuai dengan kemauan.
* Playsound (TEXT, (.wav"),NULL,SND\_ASYNC | SND\_LOOP)) untuk menjalankan lagu dalam program dengan ekstensi wav
* Penambahan gambar hanya bisa dalam format bitmap saja.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Design elemen puzzle dengan CorelDraw
* Explore input lau pada program
* Kiki Pratiwi
* Menentukan alur games
* Design karakter puzzle dengan CorelDraw
* Explore github
* Mufida Nuha Salimah
* Pemilihan warna setiap elemen dan karakter dalam RGB
* Nikita Nabila
* Menentukan karakter puzzle
* Menentukan alur games
* Input lagu
* Velia Sagita Putri
* Design layout pada kertas
* Design layout dengan IDE DevC++
* Explore trello

### Pertemuan Ke-4 (8 Maret 2018)

* **Pelaksanaan :**
* Membuat design layout dengan CorelDraw
* Menyelesaikan design layout dengan IDE DEvC++
* Membuat laporan
* Mewarnai button
* Membuat modul random elemen puzzle
* Membuat header general
* Mengedit sound
* Membuat efek tampilan awal
* Membuat UI about
* Membuat UI help
* **Pencapaian :**
* Berhasil menyelesaikan design layout pada Corel Draw
* Berhasil menyelesaikan design layout pada program
* Berhasil membuat redaksi pada game
* Berhasil membuat laporan
* Berhasil mewarnai button
* Berhasil membuat modul random elemen puzzle
* Berhsail membuat header general
* Berhasil mengedit sound.
* Berhasil membuat efek pada tampilan awal
* Berhasil membuat UI about
* Berhasil membuat UI help
* **Masalah :**
* Gagal integrase header
* Kurang terorganisirnya pembagian tugas oleh ketua
* Kesulitan menentukan waktu kerja kelompok
* **Solusi :**
* Kurang tepat syntax yg digunakan, jadi harus dipilih lagi dan diperbaiki pemahaman logicnya
* Anggota kelompok secara inisiatif mengambil bagian masing-masing menurut keahlian yang dimiliki
* Melakukan teleworking agar tugas bisa dikerjakan secara maksimal sesuai target
* **Outcome :**
* Dalam teleworking dibutuhkan koneksi internet yang stabil dan kerjasama juga kesabaran, terutama saat menunggu compile sekian lama namun ternyata masih bug
* Ukuran di CorelDraw dengan IDE DevC++ tidak sama
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Membuat design elemen
* Edit sound
* Kiki Pratiwi
* Membuat design elemen
* Membuat headear general
* Membuat Modul games
* Mufida Nuha Salimah
* Mewarnai button
* Membuat redaksi laporan
* Nikita Nabila
* Membuat laporan
* Edit sound
* Velia Sagita Putri
* Design layout dengan IDE DevC++
* Membuat modul games
* Membuat UI help
* Membuat UI about

### Pertemuan Ke-5 (15 Maret 2018)

* **Pelaksanaan :**
* Integrasi modul play
* Update mockup
* Membuat modul operasi file dalam games
* Membuat UI result
* Membuat UI exit
* Membuat modul coundowntime
* Membuat modul pause game
* Membuat modul swap
* Update modul play
* Membuat modul menampilkan level dan goal yang telah dipilih pada modul play
* Membuat laporan
* Menambahkan sound ppada tiap UI
* Update modul petak puzzle
* Update modulcoloring
* Membuat laporan
* **Pencapaian :**
* Berhasil integrasi modul play
* Berhasil mengupdate mockup
* Berhasil membuat modul operasi file
* Berhasil membuat UI result
* Berhasil membuat UI exit
* Berhasil membuat modul coundowntime
* Berhasil membuat modul menampilkan level dan goal pada modul play
* Berhasil membuat modul swap
* Berhasil mengupdate modul play
* Berhasil membuat modul pause game
* Berhail menambahkan sound pada tiap UI
* Berhasi mengupdate modul titik koordinat petak puzzle jadi menggunakan looping
* Berhasil mengupdate modul coloring
* Berhasil membuat laporan
* **Masalah :**
* Kebingungan memanggil coundowntime pada modul grafis
* Kebingungan mengatur ukuran teks yang akan ditampilkan di modul playdan result
* Bug saat integrasi modul play
* File yang disimpan tidak bisa ditampilkan
* Saat input name kedua kalinya, hasil input sebelumnya masih tersimpan
* **Solusi :**
* Saat memanggil basis teks menggunakan copystring terlebih dulu lalu ditampung ke variable, barulah pada fungsi outtextxy yang dipanggil adalah variabelnya
* Mencoba satu-satu ukuran sambil mengeksplore hasil dari tiap default dari ukuran text
* Mengubah variable global menjadi variable lokal agar tidak bug
* **Outcome :**
* Pada modular programing sangat ditekankan untuk memnggunakan variabel lokal agar tidak ada multiple declaration dan bug
* Dalam setiap modul bisa disederhanakan lagi dengan penggunaan parameter agar modul yang memiliki fungsi sama bisa dibuat 1 modul
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Update mockup
* Membuat UI input name
* Mebuat UI result
* Kiki Pratiwi
* Membuatmodul swap
* Integrasi modul play
* Membuat modul pause
* Membuat modul save data player ke file
* Mufida Nuha Salimah
* Update modul menentukan titik koordinat petak puzzle dengan looping
* Membuat Laporan
* Menyederhanakan modul coloring
* Nikita Nabila
* Membuat laporan
* Edit sound
* Membuat modul dasar operasi file
* Membuat modul exit
* Velia Sagita Putri
* Membuat laporan
* Membuat modul coundowntime
* Integrase modul play
* Membuat modul next to play
* Membuat modul menampilkan goal dan level yang telah dipilih

### Pertemuan Ke-6 (22 Maret 2018)

* **Pelaksanaan :**
* Update laporan
* Update operasi file
* Update coding menjadi clean code
* Update modul result
* **Pencapaian :**
* Berhasil mengupdate laporan
* Berhasil mengupdate operasi file
* Berhasil update coding menjadi clean code
* Berhasil update modul result
* **Masalah :**
* Kebingungan saat implementasi pengoperasian file
* **Solusi :**
* Membuka kembali materi tentang pengoperasian file, dan mengimplementasikannya pada program
* **Outcome :**
* Input nama player merupakan modul jenis fungsi yang mengembalikan tipe data string.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Update modul result
* Update coding menjadi clean code
* Kiki Pratiwi
* Update coding menjadi clean code
* Update modul operasi file
* Mufida Nuha Salimah
* Update coding menjadi clean code
* Update modul rank
* Nikita Nabila
* Update modul exit
* Update laporan dan persentasi
* Update coding menjadi clean code
* Velia Sagita Putri
  + - * Update laporan
      * Memecah prototype header gabungan kikimufidavelia.cpp
      * Update coding menjadi clean code

### Pertemuan Ke-7 ( 12 April 2018)

* **Pelaksanaan :**
* Mengubah prosedur menjadi fungsi pada beberapa modul
* Membuat konsep *custom goal*
* **Pencapaian :**
* Berhasil mengubah prosedur menjadi fungsi pada beberapa modul
* Berhasil membuat konsep kasar *custom goal*
* **Masalah :**
* Terjadi bug saat integrasi hasil perubahan modul dari prosedur menjadi fungsi
* Kebigungan menentukan konsep *custom goal* apakah dengan merging sebuah gambar atau memilih elemen satu persatu untuk diisikan ke dalam array
* **Solusi :**
* Saat memanggil fungsi, harus mengunakan variabel penampung dan harus ada nilai kembalian
* **Outcome :**
* Semuah modul fungsi harus memiliki nilai kembalian dan saat pemanggilan hasilnya harus ditampung pada sebuah variabel
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Update prosedur ke fungsi
* Membuat design UI *custom goal*
* Kiki Pratiwi
* Update prosedur ke fungsi
* Membuat konsep coding *custom goal*
* Mufida Nuha Salimah
* Update prosedur ke fungsi
* Membuat konsep coding *custom goal*
* Nikita Nabila
* Update prosedur ke fungsi
* Velia Sagita Putri
* Update prosedur ke fungsi
* Membuat konsep coding *custom goal*

### Pertemuan Ke-8 ( 19 April 2018)

* **Pelaksanaan :**
* Membuat modul *custom goal*
* Membuat coding UI *custom goal*
* Membuat gambar untuk *custom goal*
* Penyesuaian *custum goal* pada *set goal and leve*l
* Penyesuaian *random* untuk *custom goal*
* **Pencapaian :**
* Berhasil membuat modul *custom goal*
* Berhasil membuat coding UI *custom goal*
* Berhasil memilih gambar untuk *custom goal*
* **Masalah :**
* Perbedaan tipe data antara goal yang telah ditentukan sebelumnya dengan *custom goal*
* Belum bisa menampilkan *goal custom* di *screen play*
* **Solusi :**
* Ditentukan untuk mengganti tipe data disesuaikan dengan *custom goal*
* **Outcome :**
* Dengan custom cara *random* berubah menjadi lebih kompleks dan banyak kemungkinan akibat dari beragamnya kemungkinan *goal*
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Membuat gambar background untuk UI *custom goal*
* Membuat elemen untuk *custom goal*
* Kiki Pratiwi
* Membuat coding modul penempatan pilihan *goal* ke *board*
* Mufida Nuha Salimah
* Membuat coding UI *custom goal*
* Nikita Nabila
* Mencari referensi gambar elemen *custom goa*l
* Velia Sagita Putri
* Membuat modul *input custom*

### Pertemuan Ke-9 ( 26 April 2018)

* **Pelaksanaan :**
* Update *custom goal*
* **Pencapaian :**
* Berhasil memperbaiki *bug* saat *custom goal*
* Berhasil menampilkan *goal* di *screen play*
  + - * **Masalah :**
* Kesulitan mengintegrasikan dikarenakan terdapat *bug* pada modul - modul yang berkaitan dengan *custom goal*.
* Kebingungan dikarenakan *goal* yang ditampilkan tidak rapi dikarenakan koordinat gambar yang belum tepat.
* **Solusi :**
* Memahami semua alur yang berhubungan dengan *custom goal.*
* Menghitung kembali koordinat, agar bisa menampilkan *goal* di *screen play.*
* **Outcome :**
* Saat coding diperlukan ketelitian dan kesabaran yang tinggi karena sedikit kesalahan mempengaruhi program.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Update design UI *custom goal*
* Kiki Pratiwi
* Integrasi modul-modul yang berkaitan dengan input custom
* Mufida Nuha Salimah
* Update modul pemilihan gambar.
* Update modul UI *custom goal.*
* Nikita Nabila
* Update modul yang menampilkan *goal* pada  *screen play*
* Velia Sagita Putri
  + - * Update modul *input custom*

### Pertemuan Ke-10 ( 3 Mei 2018)

* **Pelaksanaan :**
* Mendiskusikan alur pembuatan *stack*
* Menerapkan struktur data *stack* untuk menyelesaikan puzzle (*solve* *puzzle*)
* **Pencapaian :**
* Berhasil menerapkan struktur data *stack*
* **Masalah :**
* Bingung menentukan subvariabel pada node yang digunakan pada struktur data *stack*
* Kesulitan saat mengimplementasikan struktur data *stack* pada game
* **Solusi :**
* Menentukan 2 subvariabel struct pada node, yaitu info pergerakan (*move*) dan info pergerakan kembalinya (*reverse*)
* Memahami kembali struktur data *stack*, setelah itu mengimplementasikan kembali struktur data *stack*
* **Outcome :**
* Stack dapat digunakan untuk mengacak (*random*) urutan puzzle dan menyelesaikan puzzle (*solve puzzle*)
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Membuat modul untuk membuat node baru
* Kiki Pratiwi
* Membuat modul yang mengubah array puzzle untuk ditampilkan saat pertama kali muncul dalam permainan
* Mufida Nuha Salimah
* Insert node ke stack dari pergerakan user
* Nikita Nabila
* Membuat modul yang menentukan batas mengacak puzzle
* Velia Sagita Putri
* Membuat modul yang mengacak urutan puzzle

### Pertemuan Ke-11 ( 17 Mei 2018)

* **Pelaksanaan :**
* Mendiskusikan bagaimana alur pembuatan *tree*
* **Pencapaian :**
* Merancang konsep bagaimana tree dibangun (*generate tree*)
* Menentukan alur bagaimana tree dibangun (*generate tree*). Tree dibangun secara level order.
* **Masalah :**
* Kebingungan penentuan kondisi pembuatan node baru yang tidak boleh sama dengan *parent*.
* Kebingungan bagaimana kondisi untuk memberhentikan pembuatan *tree*.
* **Solusi :**
* Menambah satu queue lagi untuk mengantrikan *parent*.
* Menjadikan kedalaman dari tree sebagai acuan *tree* telah selesai dibangun/dibuat.
* **Outcome :**
* Saat menmbangun *tree* berdasarkan level order diperlukan struktur data lain yaitu *queue*.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Mendiskusikan alur pembuatan *tree*.
* Kiki Pratiwi
* Mendiskusikan alur pembuatan *tree*.
* Mufida Nuha Salimah
* Mendiskusikan alur pembuatan *tree*.
* Membuat modul yang memberi info arah pergerakan puzzle.
* Mencoba membangun *tree*.
* Nikita Nabila
* Mendiskusikan alur pembuatan *tree*.
* Velia Sagita Putri
* Mendiskusikan alur pembuatan *tree*.

### Pertemuan Ke-12 ( 24 Mei 2018)

* **Pelaksanaan :**
* Menggunakan struktur data *tree* untuk mengacak urutan dan menyelesaikan puzzle
* Mencoba coding untuk membangun *tree*
* **Pencapaian :**
* Berhasil membangun *tree*
* **Masalah :**
* Kebingungan memilih subvariabel tambahan untuk mempermudah membangun *tree*, antara *index empty puzzle* atau array puzzle
* Hanya bisa membuat *tree* hingga kedalaman 7, sisanya terjadi *bug*
* **Solusi :**
* Menentukan subvariabel yang berisi info indeks yang kosong pada puzzle (*index empty puzzle*) dengan pertimbangan memori yang digunakan lebih kecil.
* Mengubah beberapa susunan algoritma
* **Outcome :**
* Dengan dibuatnya tree untuk memilih puzzle yang ditampilakan saat pertama kali bermain, menambah variasi puzzle yang teracak pada saat permainan dan tidak ada acakan yang tidak dapat diselesaikan. Dikarenakan semuanya terhubung dengan *root* (*goal* puzzle).
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Membuat modul untuk mengubah array puzzle, hasil *backtrack* pada tree
* Kiki Pratiwi
* Membuat modul untuk mengisi info node baru
* Update modul *generate tree*
* Mufida Nuha Salimah
* Membuat modul untuk mengisi info node pada root
* Membuat modul untuk perpindahan papan puzzle (*animate*) secara otomatis.
* Nikita Nabila
* Membuat modul untuk memilih node yang akan ditampilkan saat dimulai permainan
* Velia Sagita Putri
* Membuat modul untuk *backtrack* pada *tree*.
* Membuat modul untuk mengacak node yang akan ditampilkan saat dimulai permainan

### Pertemuan Ke-13 ( 31 Mei 2018)

* **Pelaksanaan :**
* Memperbaiki *bug* pada modul *generate tree*
* **Pencapaian :**
* Berhasil membangun tree
* **Masalah :**
* Kebigungan untuk mencari letak kesalahan dalam algoritma *generate tree*
* Saat membangun tree lebih dari 17 level, dibutuhkan waktu yang cukup lama untuk *generate tree*
* **Solusi :**
* Mengganti pengecekan saat membuat node baru, lalu mengubah kembali *queue* yang diguanakan yang tadinya dua queue menjadi satu *queue*.
* Menggunakan algoritma A\*, namun kami belum mampu untuk mengimplementasikannya
* **Outcome :**
* *Generate tree* pada dalam program 8 puzzle yang berdasarkan level order bisa dilakukan dengan bantuan satu *queue*.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Membuat modul yang menampilkan screen loading saat *generate tree*
* Kiki Pratiwi
* Menambahkan *controller* keyboard pada permainan
* Mufida Nuha Salimah
* Update modul *animate* puzzle
* Membuat modul untuk dealokasi *stack*
* Nikita Nabila
* Membuat modul untuk menghitung jumlah node *leaf*
* Velia Sagita Putri
* Update coding *depth* terdalam tanpa *random*

### Pertemuan Ke-14 ( 7 Juni 2018)

* **Pelaksanaan :**
* Memperbaiki *bug* di modul *animate* puzzle
* Update laporan
* Update coding menjadi clean code
* **Pencapaian :**
* Berhasil memperbaiki bug di modul *animate* puzzle
* Berhasil mengupdate laporan
* Berhasil update coding menjadi clean code
* **Masalah :**
* Kebigungan untuk mencari letak kesalahan dalam algoritma *animate puzzle*
* **Solusi :**
* Menyesuaikan kembali struktur data yang dipakai dan mengimplementasikan pada program
* **Outcome :**
* Struktur data *stack* untuk menyelesaikan puzzle tetap digunakan meskipun telah menggunakan struktur data *tree*. *Tree* digunakan untuk mengacak urutan puzzle.
* **Pembagian Tugas :**
* Ananda Khoirunisa
* Update coding menjadi clean code
* Kiki Pratiwi
* Update coding menjadi clean code
* Update modul *animate* puzzle
* Update laporan
* Mufida Nuha Salimah
* Update coding menjadi clean code
* Update laporan
* Nikita Nabila
* Update coding menjadi clean code
* Velia Sagita Putri
* Update coding menjadi clean code
* Update laporan

### 

## **Daftar Pustaka**

<http://deyanfajar.blogspot.co.id/2016/10/pembahasan-game-8-puzzle.html> Diakses pada 22 Maret 2018 jam 13.50

<https://anzdoc.com/penyelesaian-masalah-8-puzzle-dengan-algoritma-hill-climbing.html> Diakses pada 22 Maret 2018 jam 13.48

<https://www.cs.princeton.edu/courses/archive/spr10/cos226/assignments/8puzzle.html> Diakses pada 22 Maret 2018 jam 13.57

PENYELESAIAN MASALAH 8 PUZZLE DENGAN ALGORITMA HILL CLIMBING

STEPEST ASCENT LOGLIST HEURISTIK BERBASIS JAVA by azizah zakiah