Laporan Program Permainan Remi

*Dokumen Spesifikasi Program untuk Tugas Besar Mata Kuliah Dasar-Dasar Pemrograman (Praktek)*



Disusun Oleh :

Kelompok I

Ahmad Fadhitya 161511034

Fahmi Rosdiansyah Mahdani 161511044

Muhammad Hisyam Al Anshory 161511052

Rizka Fajar Novia 161511063

Kelas 1-B

**Jurusan Teknik Komputer dan Informatika**

**Program Studi D3 Teknik Informatika**

**Politeknik Negeri Bandung**

**2016**

**DAFTAR ISI**

**BAB I**

**SPESIFIKASI PROGRAM**

* 1. **Deskripsi Program**

1. **Deskripsi**

Program kelompok 1 kelas 1-B adalah mengenai permainan kartu berjenis

remi *”Rummy”* yang dimainkan oleh dua player yaitu Player 1 dan Player 2 (Komputer). Program ini dibuat dengan bahasa pemrograman C.

1. **Aturan Permainan :**
2. Masing-masing pemain memegang 7 kartu
3. Pemain bertugas membentuk deretan kartu *(meld card)* minimal 3 kartu sebagai berikut :

* *Sequence* yaitu mengurutkan kartu berjenis sama dari 2 s.d A minimal 3 kartu misalnya (3-4-5 Diamond)
* *Group* yaitu membentuk deretan kartu seri misalnya (3 CLUB - 3 HEART - 3 SPADE)
* *Dragon* yaitu mengurutkan 7 kartu dengan simbol yang sama
* *34* yaitu jika deretan kartu dibentuk dari deretan 3 kartu seri dan deretan 4 kartu seri
* Joker bisa menggantikan kartu apa saja

1. Pemain pertama mengambil kartu dari *Dek* kemudian memilih satu kartu yang tidak dikehendaki untuk dibuang di *Trash*
2. Giliran berlanjut ke pemain kedua
3. Pemain kedua mengambil kartu dari *Dek* dan memilih satu kartu yang tidak dikehendaki untuk dibuang di *Trash*
4. Giliran kembali ke pemain pertama
5. Dalam satu giliran pemain dapat mengambil kartu dari *Dek* atau dari *Trash* dan jika pemain membuang kartu Joker maka permainan berakhir
6. Permainan akan berakhir ketika ada pemain yang membuang kartu di *Dek* habis atau sudah ada pemain yang Tutup Kartu atau *Closed Card* (sudah selesai membentuk deretan kartu)
7. **Aturan Menang :**
8. Jika permainan berakhir karena ada pemain yang membuang kartu Joker maka pemain yang membuang Joker kalah, dan pemenang ditentukan dari poin tertinggi.
9. Jika permainan berakhir karena ada pemain yang Closed Card, maka pemain tersebut yang menang. Namun poin tetap dihitung.
10. Jika permainan berakhir karena kartu di *Dek* habis, maka pemenang adalah yang memiliki poin terbanyak.
11. **Aturan Poin:**

|  |  |
| --- | --- |
| Jenis kartu | Poin per kartu |
| Kartu Angka | 5 |
| Kartu Huruf | 10 |
| Kartu A | 15 |

Penghitungan poinnya adalah sebagai berikut:

1. Setiap kartu dihitung nilainya
2. Jika kartu membentuk deret maka ponnya bernilai positif (+) atau dijumlahkan
3. Jika kartu tidak membentuk deret maka poinnya bernilai negatif (-) dikurangkan
4. Jika Player menang karena Closed Card dari kartu di deck maka mendapat tambahan poin 350
5. Jika player menang karena Closed Card dari kartu di Trash maka mendapat tambahan poin sesuai kartu dan pemain yang membuang kartu tersebut mendapat minus sesuai poin yang didapat pemenang
   1. Kartu angka = 50
   2. Kartu huruf = 100
   3. Kartu A = 250
6. Jika Player menang Closed Card dan menutup menggunakan Joker maka Player mendapat tambahan 500
7. Jika sampai akhir permainan kartu joker tidak membentuk deret maka Player mendapat minus 500 poin
8. Jika ada Player yang poinnya dilewati oleh Player lain maka player yang dilewati poinnya kembali ke 0 atau (Death Match)
   1. **Identifikasi Proses**

Proses dalam permainan ini ada 3

* 1. **Tahap Awal**

Tahap awal ini berupa penginputan Nama Player kemudia dilakukan pengacakan (Suffle) kartu yang ada dan dibagikan 7 kartu pada tiap pemain

* 1. **Tahap Permainan**

Tahap dimana permainan dimulai dengan pengambilan 1 kartu di deck oleh Player 1 dan dilakukan sampai syarat untuk meme

* 1. **Tahap akhir**

Yaitu tahap penghitungan skor

* 1. **Identifikasi Data**

Data yang dibutuhkan pada program permainan ini antara lain :

1. **Dek**

Sekumpulan kartu yang belum dibagikan saat permainan belum dimulai. Dek berisi 54 kartu dengan 5 jenis kartu. Yaitu, Heart, Diamond, Club, Spade, dan Joker.

Ketentuan kartu adalah sebagai berikut

**Simbol kartu**

1 s.d 13 = HEART

14 s.d 26 = DIAMOND

27 s.d 39 = CLUB

40 s.d 52 = SPADE

53 s.d 54 = Joker (R)

**Nominal Kartu**

0 = K, 1 = A, 2 =2, 3 = 3, 4 =4, 5 =5, 6 =6, 7 = 7, 8 = 8, 9 = 9, 10 = 10, 11 = J, 12 = Q

1. **Trash**

Array tempat menampung kartu yang dibuang oleh Player.

1. **Dek Player**

Dek Player adalah sekumpulan kartu yang dimiliki Player setelah kartu dari dek dibagikan.

1. **Point atau Score**

Point yaitu variabel penampung nilai dari setiap Player setelah permainan selesai

**BAB II**

**PERANCANGAN PROGRAM**

**2.1 Perancangan Proses**

1. **Structured Chart**

Modul Main()

Didalamnya terdapat pemanggil modul:

1. StartGame()

2. howToPlay()

3. exit()

HowToPlay()

gameInit()

update()

if(now.PlayerNumber==1)

gameboard(&player1, &player2)

if (player1.Status==0)

startAI(&player1)

else

playerCommand(&player1)

endif

else(now.PlayerNumber==2)

gameboard(&player2, &player1)

if (player2.Status==0)

startAI(&player2)

else

playerCommand(&player2)

endif

endif

player

setPlayerScore(player)

setPlayerName(player)

setPlayerStatus(player)

player

rouindInit()

player

resetDeck()

shuffleDeck()

emptyTrash()

emptyPlayerDeck(&player1)

emptyMeldDeck(&player1)

dealFromDeck(&player1)

emptyPlayerDeck(&player2)

emptyMeldDeck(&player2)

dealFromDeck(&player2)

playerCommand(\*player)

if now.Turn==1

option1(&player)

else now.Turn==2

option2(&player)

else now.Turn==3

option3(&player)

1. **Penjelasan Global Modul**
2. **Algoritma**

**//Kamus Data**

int deckLengthNow = 54

int trashLengthNow = 0

int deck[DECK\_CARD\_LENGTH]

int godMode = 0

int debugMode = 0

int difficulty

**Pembuatan Type Data Record**

* playerControl **variabel bertype record** dengan subvar:

playerName[100] **var tunggal bertipe array of char**

int card, meldLength, melded, cardLength, score, status, **var tunggal bertipe integer**

meldCard[PLAYER\_CARD\_LENGTH] **var array bertipe integer**

* gameControl **variabel bertype record** dengan subvar:

playerNumber, turn, difficulty **var tunggal bertipe integer**

**Modul Utama**

Main Modul()

Begin

call modul menuDisplay()

switch(keyboard):

case 1: call modul StartGame()

case 2: call modul HowToPlay()

case 3: exit()

end Main Modul

**Modul StartGame**

procedure startGame()

begin

call modul gameInit()

call modul update()

end procedure

**Modul gameInit**

procedure gameInit()

begin

call modul setPlayerScore(&player1, 0);

call modul setPlayerScore(&player2, 0);

call modul getPlayerName();

call modul getPlayerStatus();

call modul roundInit()

end procedure

**Modul roundInit**

procedure roundInit()

begin

call modul resetDeck();

call modul shuffleDeck();

call modul emptyTrash();

call modul emptyPlayerDeck(&player1);

call modul emptyMeldDeck(&player1);

call modul dealFromDeck(&player1);

player1.melded 0;

call modul emptyPlayerDeck(&player2);

call modul emptyMeldDeck(&player2);

call modul dealFromDeck(&player2);

player2.melded 0;

now.playerNumber 1;

now.turn 1;

end procedure

**Modul resetDeck**

procedure resetDeck()

//i var lokal type integer

deckLengthNow DECK\_CARD\_LENGTH

i 0

for (i<DECK\_CARD\_LENGTH)

deck[i]=i

i i + 1

end for

end procedure

**Modul shuffleDeck**

procedure shuffleDeck()

i var lokal bertipe integer

random(time(NULL))

i DECK\_CARD\_LENGTH-1

for (i>0) {

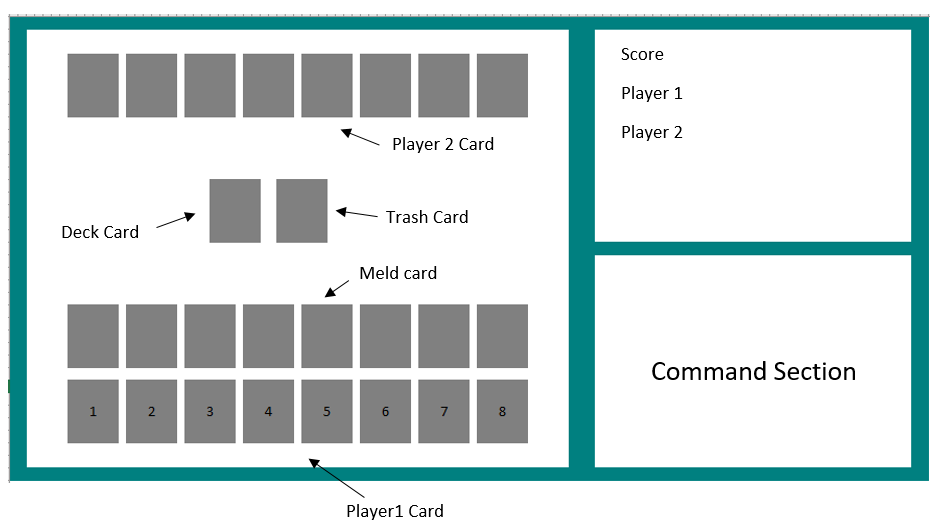
int j = rand() % (i+1)

i i - 7

call modul swapCard(deck, i, j);

**2.2 Perancangan Tampilan (Output)**

**Welcome Screen UI**

**Gameplay UI**

**BAB III**

**KESIMPULAN**