



## *SSC0600 – Introdução à Ciência de Computação I*

Prof. Dr. Adenilso da Silva Simão



Nome: Paulo Inay Cruz (p = 0)

Nº USP 10388691

Nome: Jonathan Ferreira de Mello (p = 1)

Nº USP 10377754

Nome: João Marcos Della Torre Divino (p = 2)

Nº USP 10377708

### **RELATÓRIO – TRABALHO 4 – RUMMIKUB**

#### **Introdução**

Neste trabalho, foi implementado, em linguagem C, um programa que permite jogar Rummikub, que consisti num jogo de mesa composto por 106 peças (neste caso, cartas), sendo 104 numeradas (de 1 a 13) e divididas em cores (neste caso, naipes), e dois coringas. Cada jogador recebe/escolhe 14 peças, e na sua vez, objetiva esvaziar sua mão ao combinar na mesa/tabuleiro as peças entre si, formando grupos ou sequências (de no mínimo 3 peças). Grupos são formados por peças de mesmo valor, mas naipes diferentes, enquanto sequências são formadas por peças do mesmo naipe, indo de 1 a 13. Ganha quem colocar todas as peças primeiro no tabuleiro.

Para o trabalho, as regras foram um pouco simplificadas, porém sem ferir os princípios do jogo. O baralho foi composto por números hexadecimais (de 1 a D) e por caracteres especiais para a identificação dos naipes ('!', '@', '#' e '\$'), sendo os coringas representados por '\*'.

O programa em si fornece uma interface interativa e bem detalhada, apresentando um menu inicial, opção para o modo de jogo e para a quantidade de jogadores, um menu secundário com as ações do jogo, visão do tabuleiro e das cartas de cada jogador, indicação de vitória e possibilidade de jogar novamente.

## Descrição do projeto

Este trabalho foi desenvolvido em dois computadores, ambos com sistema operacional Windows 10 64-bits. Quanto ao ambiente de programação, foram utilizados o programa Dev-C++ 5.11, com o compilador TDM-GCC 4.9.2 64-bit, e o Code::Blocks 17.12, com o compilador GNU GCC (MinGW) x64.

O código fonte é “Rummikub.c” e os headers utilizado foram “stdio.h”, “stdlib.h”, “string.h” e “time.h”. A estrutura é basicamente composta por funções, que controlam o andamento do jogo, e por menus cuja entrada é, em sua maioria, composta por caracteres. Além disso, o papel de cada variável e de cada função está explicitado no código na forma de comentários, a fim de deixar claro o que se está fazendo e manter a organização do código.

## Tutorial

A compilação do programa dependerá do sistema operacional instalado no computador, além do ambiente de desenvolvimento.

Para usuários Windows, pode-se realizar a compilação no Dev-C++, no CodeBlocks, ou em outro software de escolha. Escolhendo o Dev-C++, após instalá-lo, basta abri-lo, selecionar a opção “File” e, em seguida, “Open” (Imagem 1).

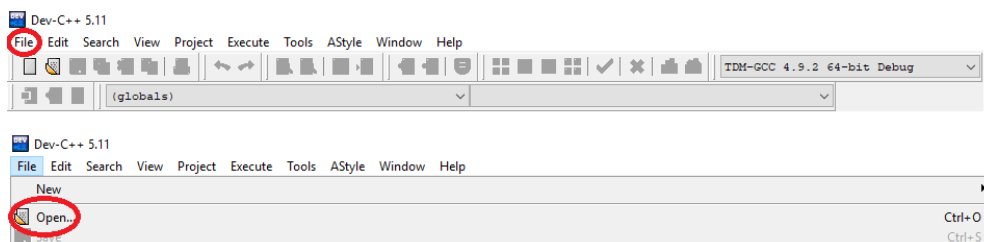


Imagem 1 – Abrindo o código no Dev-C++.

Então, busca-se e seleciona-se o arquivo “Rummikub.c” e, por fim, com ele já aberto, clica-se no botão “Compile & Run” para executar o código (Imagem 2).

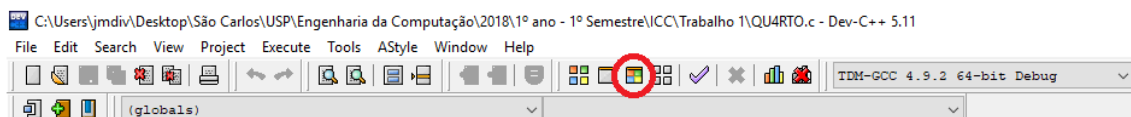


Imagem 2 – Compilando e executando o código.

Caso opte-se pelo CodeBlocks, o procedimento é análogo. Com o programa instalado, seleciona-se a opção “File” e, depois, “Open” (Imagem 3).

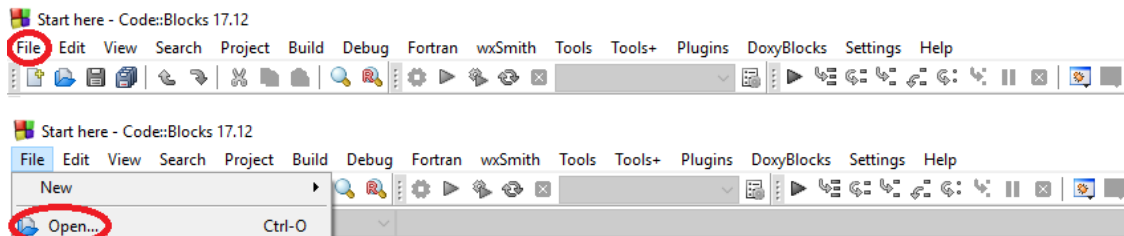


Imagem 3 – Abrindo o código no CodeBlocks.

Então, abre-se o código “Rummikub.c” e seleciona-se “Build and run” para compilar e rodar (Imagem 4).

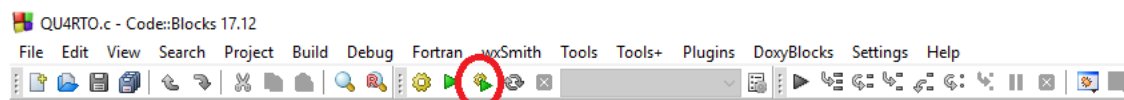


Imagem 4 – Compilando e executando o código.

Já para usuários Linux, com o GCC instalado, basta executar o seguinte comando no terminal para compilar o código: `gcc Rummikub.c -o Rummikub`. Para executá-lo, digite o comando `./Rummikub`.

A execução do programa, porém, independe dos elementos citados.

Assim que o programa executar, abrirá uma janela que apresentará a tela inicial do jogo (Imagem 5). Nela o usuário poderá seguir duas opções: “Jogar” ou “Sair”. Escolhendo sair, o programa é encerrado.

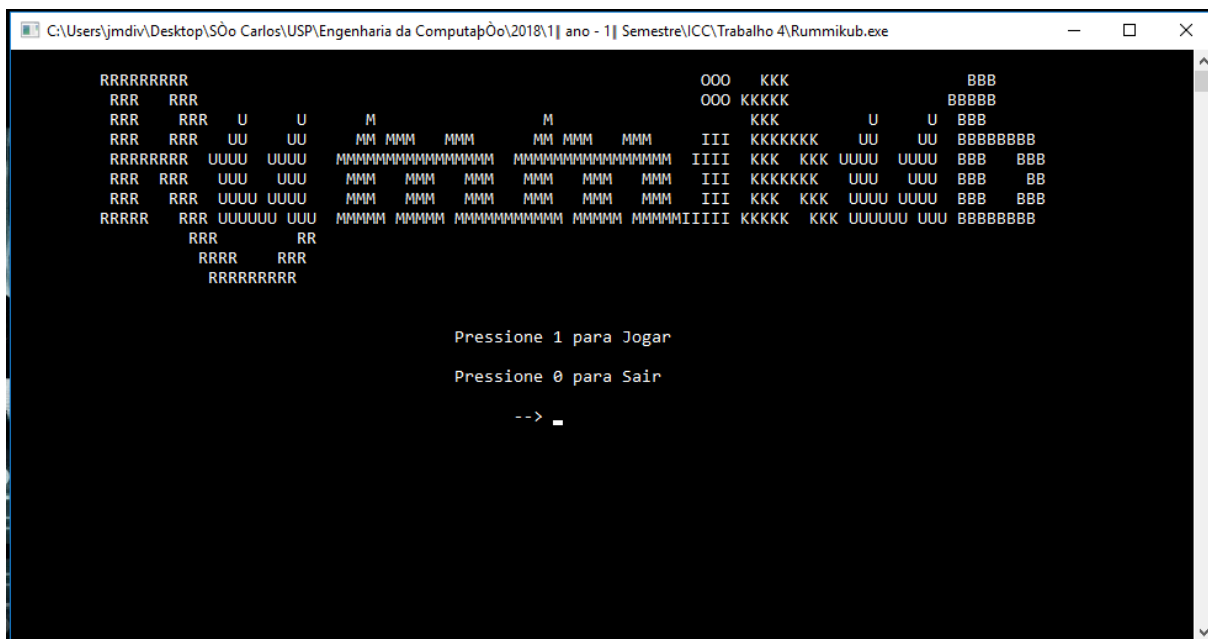


Imagem 5 – Menu inicial do programa.

Já se optar por “Jogar”, ele é conduzido a uma nova tela, na qual irá selecionar o modo de jogo e a quantidade de jogadores que participarão da partida (Imagem 6). No modo aleatório, o baralho é gerado desordenadamente, enquanto no modo controlado, ele é lido de um arquivo “.txt”.

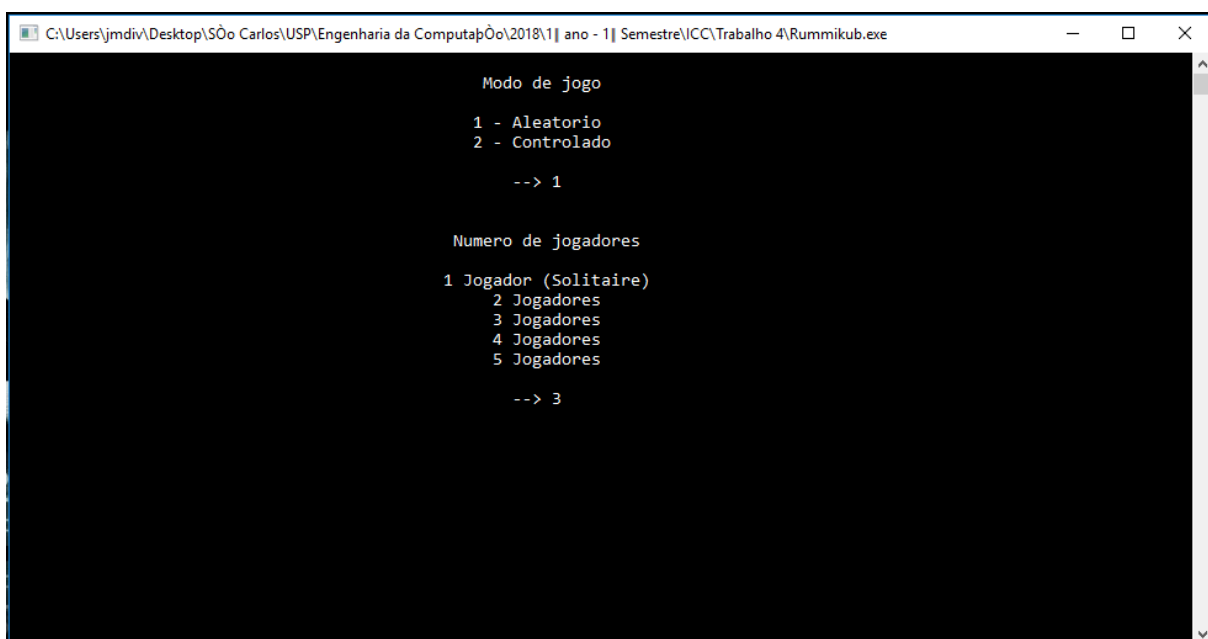


Imagem 6 – Tela de configuração do jogo.

Após configurar a partida, esta tem início, sendo mostrado ao usuário o tabuleiro do jogo, as cartas de cada jogador, qual é o jogador da vez, e um menu de opções a serem realizadas (Imagem 7).

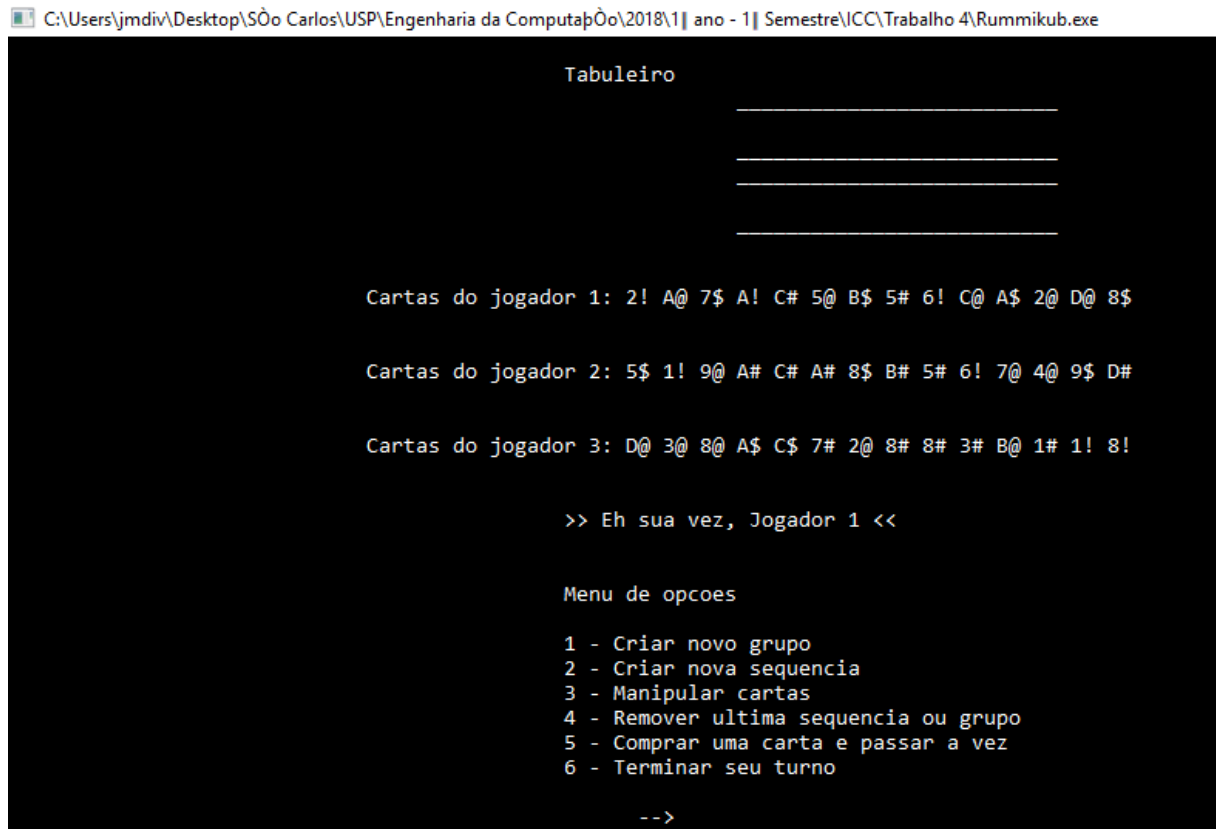


Imagem 7 – Exemplo da tela do jogo propriamente, para uma partida com 3 jogadores.

Na sua vez, o jogador poderá escolher entre “Criar novo grupo”, fazendo com que um grupo seja adicionada ao tabuleiro (Imagem 8). Processo análogo ocorre com a seleção da opção “Criar nova sequência”.

Optando por manipular cartas, o usuário é levado a um menu, no qual primeiro é questionado sobre onde se encontra a carta. Se esta estiver em sua mão, ele deve inserir o conteúdo da carta, e decidir se colocará em uma sequência ou grupo (Imagem 9 e 10). Se a carta estiver num grupo, ele deve indicar qual é a carta, em qual grupo está, e o que será feito. Se estiver numa sequência, é requerido indicar a sequência, a posição da carta na sequência e o que será feito.

C:\Users\jmdiv\Desktop\SÔo Carlos\USP\Engenharia da Computação\2018\1º ano - 1º Semestre\ICC\Trabalho 4\Rummikub.exe

```
Tabuleiro
_____

Grupo 1
_____
_____
_____

Cartas do jogador 1: 2! A@ 7$ A! C# 5@ B$ 5# 6! C@ A$ 2@ D@ 8$

Cartas do jogador 2: 5$ 1! 9@ A# C# A# 8$ B# 5# 6! 7@ 4@ 9$ D#

Cartas do jogador 3: D@ 3@ 8@ A$ C$ 7# 2@ 8# 8# 3# B@ 1# 1! 8!

>> Eh sua vez, Jogador 1 <<

Menu de opcoes

1 - Criar novo grupo
2 - Criar nova sequencia
3 - Manipular cartas
4 - Remover ultima sequencia ou grupo
5 - Comprar uma carta e passar a vez
6 - Terminar seu turno

-->
```

Imagem 8 – Adição de um grupo ao tabuleiro.

C:\Users\jmdiv\Desktop\SÔo Carlos\USP\Engenharia da Computação\2018\1º ano - 1º Semestre\ICC\Trabalho 4\Rummikub.exe

```
Menu de opcoes

1 - Criar novo grupo
2 - Criar nova sequencia
3 - Manipular cartas
4 - Remover ultima sequencia ou grupo
5 - Comprar uma carta e passar a vez
6 - Terminar seu turno

--> 3

Selecionar carta

1 - Mao
2 - Sequencia
3 - Grupo

--> 1

Insira a carta (numero e naipe) --> A@

Posicionar carta

1 - Sequencia
2 - Grupo

--> 2

Em qual grupo deseja colocar a carta?

--> 1

Carta adicionada com sucesso!!!
Deseja mover mais cartas?

1 - Sim
2 - Nao

-->
```

Imagem 9 – Passos após seleção da opção 3 - Manipular cartas.

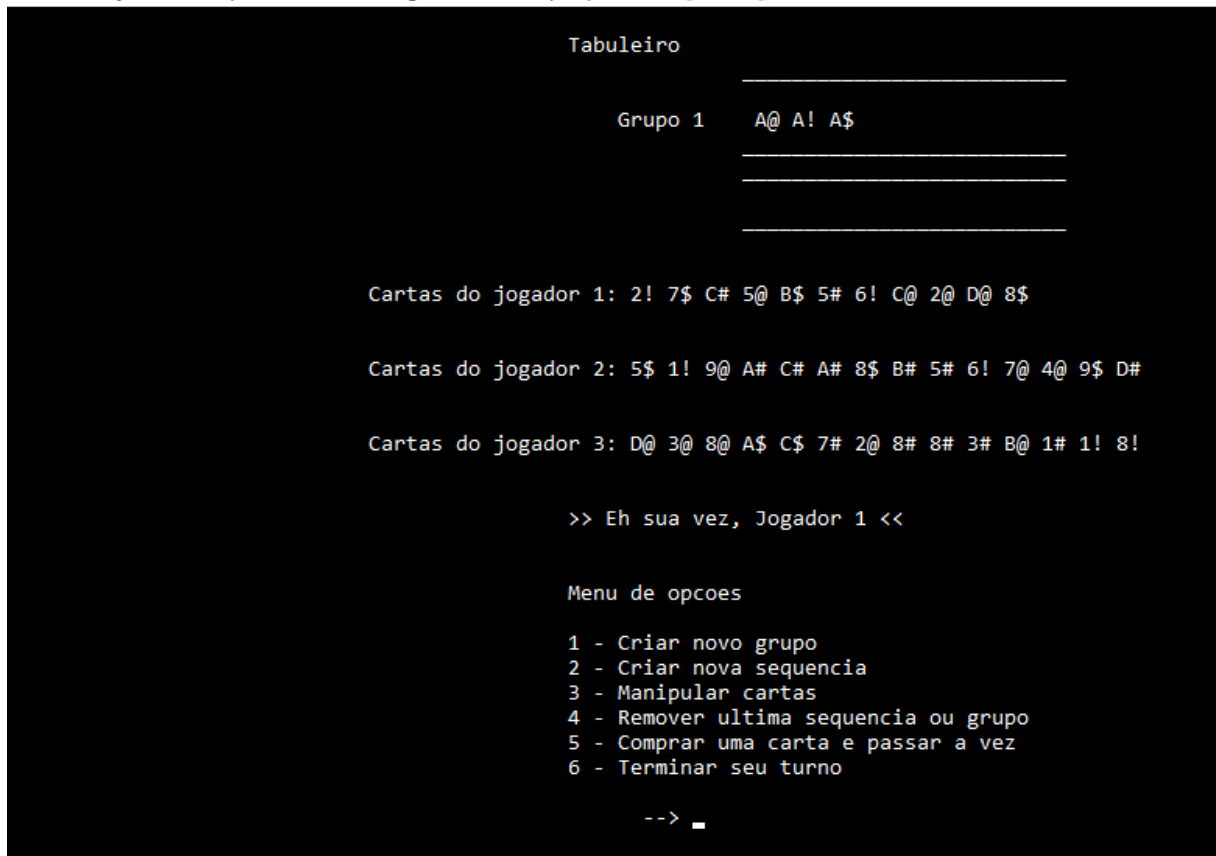


Imagem 10 – Tela de jogo após a adição de um grupo ao tabuleiro.

Caso queira remover um grupo ou sequência do tabuleiro, o jogador deve selecionar a opção 4. Se for necessário comprar uma carta, a opção 5 deve ser selecionada, sendo passada a vez logo em seguida. Por fim, após terminar suas ações, a opção 6 finaliza o turno e dirige as ações ao próximo jogador.

É importante mencionar que cada movimento é verificado, a fim de impedir qualquer descumprimento das regras do jogo.

Assim, o jogo prossegue até que seja detectado um vencedor, que acontecesse caso algum jogador não tenha mais cartas em sua mão (Imagem 11). Outra condição que encerra a partida é acabarem as cartas para serem compradas. Neste caso, cada jogador tem sua mão verificada e aquele que apresentar a menor soma dos valores das suas cartas, vence. Por fim, o usuário pode escolher entre jogar novamente, ou encerrar o jogo.

```

Tabuleiro
_____

Grupo 1    1! 1# 1$ 1@
Grupo 2    2! 2# 2$ 2@
Grupo 3    3! 3# 3$ 3@
Grupo 4    4! 4# 4$

_____
_____
_____

Cartas do jogador 1:

Cartas do jogador 2: 1! 1# 1$ 1@ 2! 2# 2$ 2@ 3! 3# 3$ 3@ 4! 4# 4$

>> Eh sua vez, Jogador 1 <<

Menu de opcoes

1 - Criar novo grupo
2 - Criar nova sequencia
3 - Manipular cartas
4 - Remover ultima sequencia ou grupo
5 - Comprar uma carta e passar a vez
6 - Terminar seu turno

--> 6

Jogador 1 venceu!!!

Deseja jogar novamente?

Pressione 1 para Sim
Pressione 0 para Nao

-->
```

Imagem 11 – Tela final do jogo após indicação de vitória.

## Bugs e limitações

O programa apresenta limitações no que diz respeito à alguns menus que trabalham com entradas int. Caso seja digitado algo diferente, o menu entra em loop infinito. Além disso, aleatoriamente, é imprimido nas cartas da mão de algum jogador os caracteres “C:”, quando o anterior compra uma carta e passa a vez.