

# **Trabalho Final**

## **Jogo de Uno**

grupo [tb-19](#):

**Matheus Schimieguel Silva**

**Helena Pato Magalhães**

**Tiago Tozo**

## Regras adaptadas do Uno

O jogo consiste em um baralho específico para jogá-lo. Cada carta tem uma cor e um valor. Existem 5 cores: azul, verde, vermelho, amarelo e especial (preta) e 15 valores que são pular, reverter, +2, coringa e +4, além dos valores numéricos de 0 a 9. As cartas de cor preta são apenas as cartas coringa e +4, de modo que esses valores não podem ser de cor azul, verde, vermelho ou amarelo. São 108 cartas distribuídas da seguinte forma :

19 cartas azuis - 0 a 9

19 cartas verdes - 0 a 9

19 cartas vermelhas - 0 a 9

19 cartas amarelas - 0 a 9

8 cartas +2 : 2 de cada cor (azul, verde, vermelho amarelo)

8 cartas pular : 2 de cada cor (azul, verde, vermelho amarelo)

8 cartas reverter : 2 de cada cor (azul, verde, vermelho amarelo)

4 cartas coringas : cor preta

4 cartas +4 : cor preta

O objetivo do jogo consiste em acabar com as cartas da sua mão antes dos adversários. Uma carta só pode ser jogada se ela tiver a mesma cor ou valor da carta no topo da pilha onde os jogadores jogam suas cartas, ou ainda se a carta for de cor especial (preta). A cada rodada o jogador atual ou joga uma carta ou compra uma, caso a carta comprada seja jogável ele pode jogá-la.

As cartas com valores diferentes de valores entre 0 e 9 possuem efeitos ao serem jogadas, são eles:

### **+2 :**

Quando esta carta é jogada, o próximo comprará 2 cartas ou, caso ele jogue outro +2, o próximo jogador terá comprar 4 cartas. Caso o último, por sua vez, jogue mais uma carta +2 o próximo jogador terá que comprar 6 cartas e assim por diante.

### **reverter :**

Reverte o sentido do jogo. Quando há apenas dois jogadores jogando, o reverter tem mesmo efeito da carta pular.

### **pular :**

O próximo jogador é pulado.

#### **+4 :**

Quando esta carta é jogada, o próximo jogador comprará 4 cartas, ou caso ele também jogue +4 o seguinte terá que comprar 8 cartas. Caso o último, por sua vez, jogue uma carta +4 o próximo jogador terá que comprar 12 cartas e assim por diante. O jogador que jogar o último +4 da corrente pode escolher a cor da próxima carta a ser jogada.

## **Especificações do código**

### **classes**

#### **Carta:**

A carta tem duas variáveis do tipo char que representam a cor e o valor além de um int, que guarda a posição do vector do jogador/bot alvo, para identificar o alvo da carta caso ela tenha um efeito, de modo a tornar mais simples a sua representação. O código foi pensado para que todas as cartas fossem alocadas no início do jogo e que nenhuma carta fosse destruída até final do código.

#### **Baralho:**

O baralho é um vector de ponteiros para Carta sendo que seu construtor, sem parâmetros, aloca no heap as 108 cartas (que foram descritas) que serão usadas durante o jogo. Durante a execução do jogo nenhuma carta é alocada, apenas no início dele.

#### **Jogador:**

A classe Jogador possui uma string como nome e um vector que guarda as cartas que aquele jogador tem. Os ponteiros das cartas são tirados das mãos de um jogador e passado para a pilha, de modo que a carta não seja destruída. Os métodos que exigem a entrada do jogador estão todos na classe jogador de modo que, por meio de polimorfismo, o Bot possa executar as mesmas ações por meio dos mesmos métodos porém com instruções mais adequadas.

#### **Bot:**

A classe Bot é uma classe filha de Jogador que faz uso de polimorfismo, de modo que o código, que havia sido feito inicialmente pensando na classe Jogador, não sofresse modificações na classe jogo e no

método rodada. O bot possui uma variável static int que representa o número de bots existente de modo que dois bots não podem ter um mesmo nome. Os bots possuem um ponteiro constante para o jogo o qual fazem parte de modo que eles possam acessar informações que lhes permitam computar a jogada adequada .Obs: o bot possui acesso apenas às informações que os outros jogadores têm, como o número de cartas dos outros jogadores, o sentido do jogo, o próximo jogador, etc.

### **Jogo:**

A classe Jogo é a classe responsável por agregar todas as outras classes como atributos e fazer a mecânica do jogo uno. Possui um ponteiro de Baralho, um vector de ponteiros para Jogadores, uma list de ponteiros para cartas (que representa a pilha onde os jogadores vão jogar suas cartas), assim como int para guardar a quantidade de bots e de jogadores, além do sentido do jogo.

O método rodada é o responsável por fazer o ciclo do jogo representando a vez de cada jogador, caso o jogador não seja pulado.

### **Mecânicas :**

#### **Jogada:**

Por meio de um método de jogador é possível saber se um jogador tem cartas que possam ser jogadas, caso verdadeiro ele pode jogar uma carta de sua mão que seja jogável, isso se ele não for alvo de um +2 ou +4. Caso o jogador não tenha cartas jogáveis ele é obrigado a comprar uma carta. Caso ele seja alvo de um +2 ou +4, jogador chama um método de jogada específico para essa situação: o rebate. O bot possui uma sequência de preferências para jogar uma carta.

#### **Rebate:**

Cartas como +2 e +4 ao serem jogadas a um determinado jogador podem ser rebatidas fazendo com que o próximo jogador sofra as consequências da corrente de +2 ou +4. Por exemplo, se há uma sequência de quatro +2 o jogador que sofrer a ação da última carta jogada terá que comprar 8 cartas. Assim que o jogador comprar as cartas os alvos dessa sequência de cartas são setados para -1. Um jogador pode escolher se quer

rebater uma sequência de +2/+4 ou não, entretanto um bot sempre escolhe por se defender.

#### **escolher cor:**

As cartas de cor especial (preta), +4 e coringa, permitem ao jogador mudar a cor dessa carta que está na pilha para verde, vermelho, azul ou amarelo. Assim na pilha não podem existir cartas com cor especial pois essa é mudada assim que a carta é jogada.

#### **reverter:**

Apenas muda o sentido do jogo, multiplicando um inteiro por menos um. No construtor do jogo ele é setado para 1 e portanto em todo código só podem assumir valores 1 e -1. Quando há apenas 2 jogadores reverter tem a mesma mecânica e efeito de pular um jogador.

#### **pular:**

Ao pular um jogador o cálculo para determinar o próximo jogador é feito duas vezes.

#### **+2 ou +4:**

Toda vez que um jogador sofre um efeito de uma dessas cartas ou de uma sequência dessa ele compra duas ou quatro vezes o tamanho da sequência de que foi alvo e assim que compra também é pulado. Obs: um +2 não rebate um +4 tão pouco o contrário.

#### **vai jogar:**

Vai jogar é um método que pergunta ao jogador se ele quer jogar ou se ele prefere comprar uma carta e passar a vez. Para o bot ele sempre opta por jogar uma carta.

#### **comprar carta:**

Um jogador ou um bot sofre o mesmo efeito de comprar, que é um método que necessita da passagem de um baralho por referência. Sempre que um jogador compra uma carta em razão de não possuir uma carta jogável, ou em razão de sua escolha de não querer jogar as cartas que tinha, ele pode jogar a carta comprada, se ela for jogável. Isso é feito chamando o método rodada novamente.

Jogada, vai jogar e escolher cor são métodos polimórficos, possuindo ações diferentes para bots e para jogadores.

**Referências:**

Treinamento em linguagem C++ 2 edição Victorine Viviane Mizrahi  
site cplusplus: <http://www.cplusplus.com/reference/>  
slides PDS II.