

## MAC0110 – Introdução à Computação

Primeiro semestre de 2019

### Quebrando a banca – Entrega: 28 de abril de 2019

**Importante:** Este exercício-programa é individual. Não compartilhe sua solução com outros colegas da sala. Você pode conversar sobre o EP, dar dicas, mas, **nunca** empreste seu código. Fazer os EPs é parte fundamental da disciplina.

**Você deve utilizar somente os comandos e tipos de variáveis vistos em sala de aula.**

Neste EP vocês farão um programa em C que implementará um gerador de números aleatórios, e utilizarão este gerador para achar uma boa estratégia para jogar 21 (*blackjack*).

### Gerando números pseudo-aleatórios

A primeira tarefa neste EP será construir um gerador de números pseudo-aleatórios. Seu gerador gerará números no intervalo  $(0, 1)$ , conforme a estratégia descrita a seguir.

Dado um número  $x$  entre 0 e 1, chamado de **semente** (por exemplo, no início podemos começar com o número 0.<seu número usp>), você pode gerar o próximo número da sequência através da operação:

```
x = frac(c1 * modulo(seno(x)) + c2);
```

onde **frac** é uma função que devolve a parte fracionária de um número real, **modulo** é uma função que devolve o valor absoluto de um número real e **seno** é uma função que devolve o valor do seno de um número real. As constantes  $c1$  e  $c2$  são dois números reais quaisquer (teste várias escolhas).

Para calcular os valores de **seno(x)** você deve usar OBRIGATORIAMENTE a aproximação da função por sua expansão em série de Taylor:

$$\text{seno}(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + \frac{(-1)^k x^{2k+1}}{(2k+1)!} + \dots$$

incluindo, por exemplo, todos os termos até que  $\frac{|x^{2k+1}|}{(2k+1)!} \leq 10^{-8}$ .

## O jogo de 21

O 21, também chamado de *blackjack* é um jogo muito famoso em cassinos. É jogado com cartas usuais do baralho. As cartas de 2 a 10 contam tantos pontos quanto seu valor. O ás conta 1 ponto, e as figuras (J, Q e K) contam 10 pontos. O objetivo do jogo é comprar cartas de forma que a soma dos pontos correspondentes às cartas seja tão próxima de 21 quanto possível. O jogador ganha se fizer mais pontos que a banca, desde que não ultrapasse 21. A banca joga depois que o jogador parou de pegar cartas.

Exemplo:

- O jogador tem um rei e um 7 (17 pontos). A banca, então, pega um 4, um 5 e um valete (19 pontos). Ganha a banca.
- O jogador tem um ás, um 8 e um 7. A banca pega uma dama e um 5. Ganha o jogador.
- O jogador tem um valete, um rei e um ás. Fez 21 pontos, ganhou.
- O jogador tem um 8 e um rei. A banca pega um 7, um rei e um 8. Estourou, ganha o jogador.
- O jogador tem um valete, um 2 e um rei. Estourou, ganha a banca.

## Qual a melhor estratégia do jogador?

Quando você está jogando 21 sempre fica na dúvida se deve ou não comprar uma nova carta. Se o fizer, pode estourar e perder. Mas, se não estourar, pode ficar com uma pontuação melhor, e terá mais chances de ganhar. A ideia deste EP é testar vários valores possíveis para limiar de sua estratégia. Sempre que o total de pontos na sua mão atingir o limiar, você para e não pede mais cartas.

Já a banca tem uma estratégia muito simples. Como ela sabe quantos pontos você tem, ela segue pegando cartas até que, ou ela estoura ou ela atinge uma pontuação vencedora.

Para cada limiar possível (12, 13, ..., 20) você deverá simular vários jogos, aplicando a estratégia do jogador e da banca, e contabilizar quantos destes jogos você ganhou com cada limiar. Com isso você poderá ter uma boa ideia de qual estratégia usar na sua próxima visita a Las Vegas, e também quais são suas reais chances usando esta estratégia.

## Especificação do programa

Neste exercício-programa você deverá escrever um programa em C que testa vários limiares como estratégia para o jogo de 21. Para cada um dos limiares (de 12 a 20), você deverá executar várias partidas aleatórias. A estratégia do jogador será pegar cartas até atingir o limiar. A banca joga em seguida, e tenta ganhar do jogador. Se ultrapassar a pontuação do jogador sem estourar, a banca ganha. Caso contrário, o jogador ganha.

Você deve imprimir, para cada limiar, o percentual de jogos ganhos e asteriscos (“\*”) em número proporcional à porcentagem de jogos ganhos. Com isso poderá avaliar qual o melhor limiar.

Vamos sortear as cartas sem naipes. Assim, as cartas serão números inteiros entre 1 e 13. Utilizando a ideia acima, que gera um número  $x$  real no intervalo  $(0, 1)$  podemos gerar uma carta (inteiro em  $[1, 13]$ ) fazendo:

```
carta = 13 * x + 1;
```

A semente deverá ser lida no início do programa. Lembre que é um número real no intervalo  $(0,1)$ . Também devemos ler quantos jogos deverão ser simulados.

## Exemplo de execução do programa

```
Digite a semente (0 < x < 1): .3267962
```

```
Digite o número de simulações para cada limiar: 10000
```

```
12 ( 29.1%) : *****
13 ( 33.1%) : *****
14 ( 36.2%) : *****
15 ( 38.5%) : *****
16 ( 38.1%) : *****
17 ( 37.6%) : *****
18 ( 35.0%) : *****
19 ( 29.2%) : *****
20 ( 23.5%) : *****
```