

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO – DEPARTAMENTO DE INFORMÁTICA

Trabalho Computacional – Programação Básica de Computadores – 2018/2

Data para entrega: 30/11/2018

Uma casa de recreação de idosos, “**Sem Censura**”, está com um projeto de inserção no mundo digital. Para iniciar tal projeto, deseja que você desenvolva um programa para dois jogos. Cada jogo deve ter a participação de, no máximo, 10 (dez) jogadores.

O seu programa deve, primeiramente, ler o número de jogadores, em seguida, ler um código **1** ou **2**, que define o tipo de jogo. É obrigatório construir uma função para o **JOGO 1** e uma função para o **JOGO 2**. Abaixo seguem as descrições de cada jogo.

JOGO 1: jogo dos 30 aleatórios

- cada jogador fornece o seu nome;
- cada jogador fornece o valor de sua aposta no intervalo de 5 até 20 doces sortidos;
- calcular o valor do prêmio que é a soma de todas as apostas dos jogadores;
- para cada jogador, o computador gera 30 números aleatórios no intervalo [1,1000];
- calcular a soma **S** desses 30 números;
- o jogador que obtiver a maior soma **S** é o vencedor;
- em caso de empate, o jogador cujos números possuírem o menor **desvio padrão** é o vencedor;
- o vencedor receberá o valor total do prêmio;
- se o empate persistir, o prêmio total é dividido pelo número de jogadores empatados. Caso a divisão não seja inteira, o resto dos doces é consumido pelos criadores dos jogos.

Para o **JOGO 1**, seu programa deve conter, pelo menos, as seguintes funções: (a) gerar os 30 números aleatórios no intervalo determinado, (b) calcular a soma **S**, (c) calcular o desvio padrão através da fórmula **Desvio** = $\sqrt{1/(n-1) \sum (x_i - \bar{x})^2}$ considerando n amostras x_i e a respectiva média \bar{x} .

Como saída deste jogo, imprimir em arquivo chamado “**jogo1.txt**”, um relatório da partida que conterà, para cada jogador, seu nome, o valor da aposta, os 30 números gerados e o valor de **S**. Além disso, imprimir o nome do vencedor e o valor do prêmio total. Em caso de empate da soma **S**, imprimir o **desvio padrão** calculado para cada jogador que obteve o mesmo valor de **S** e o nome do vencedor com o valor do prêmio total. Se o empate persistir, imprimir os nomes dos empatados e o valor do prêmio que cada um receberá.

JOGO 2: jogo do bingo de cartela única

- cada jogador fornece o seu nome;
- cada jogador fornece o valor de sua aposta no intervalo de 5 até 20 doces sortidos;
- calcular o valor do prêmio que é a soma de todas as apostas dos jogadores;
- o computador gera a cartela que contém 30 números aleatórios no intervalo [1,1000]. Se houver números repetidos, atribuir valor 0 (zero) aos valores repetidos, deixando apenas o primeiro valor repetido. **Exemplo:** 21 43 21 21 90 998 90 → 21 43 0 0 90 998 0;
- para cada jogador, gerar vários números aleatórios, no mesmo intervalo de [1,1000], até que todos os números diferentes de 0 (zero) da cartela única sejam sorteados;

- o jogador que necessitar de menor número de gerações aleatórias até completar a cartela é o vencedor e receberá o valor total do prêmio;
- se houver empate, o prêmio total é dividido pelo número de jogadores empatados. Caso a divisão não seja inteira, o resto dos doces é consumido pelos criadores dos jogos.

Para o **JOGO 2**, seu programa deve conter, pelo menos, as seguintes funções: (a) gerar a cartela com os 30 números no intervalo determinado, zerando os valores repetidos, como no exemplo acima e (b) gerar os números aleatórios até completar a cartela, retornando o valor da quantidade de números gerados.

Como saída deste jogo, imprimir em arquivo chamado “**jogo2.txt**”, um relatório da partida que conterá a cartela única do bingo e, para cada jogador, o nome, o valor da aposta. Além disso, imprimir o nome do vencedor, número de gerações de aleatórios até sortear todos os valores da cartela e o valor do prêmio total. Em caso de empate no número de gerações até sortear todos os valores da cartela, imprimir os nomes dos empatados, o número de gerações até sortear todos os valores da cartela e o valor do prêmio que cada um receberá.

Sobre a estrutura de dados:

- No **JOGO 1**, o aluno deve criar, usando a função **malloc(.)**, um vetor de structs com os campos: nome, valor da aposta, vetor com os 30 números gerados, o valor **S** (soma) e o valor do **desvio padrão**.
- No **JOGO 2**, o aluno deve criar, usando a função **malloc(.)**, um vetor de structs com os campos: nome, valor da aposta, número de gerações até sortear todos os valores da cartela.

Importante: Além de todas as funções criadas e organizadas adequadamente pelo aluno, é obrigatório utilizar uma função para ordenação. O algoritmo de ordenação e a forma de utilização dessa função no trabalho são definidos pelo aluno.

Observações sobre a entrega:

- O trabalho pode ser feito em grupo de no máximo 2 alunos. Colocar no cabeçalho do trabalho os **nomes completos** dos componentes do grupo.
- O grupo deve entregar o trabalho até 30/11/2018 via e-mail para crangel@inf.ufes.br até 23h59min. Anexar os arquivos: trabalho.c (não salvar com extensão cpp) e as bibliotecas organizadas.
- Os grupos que, por ventura, atrasarem a entrega terão 0,5 ponto descontado para cada dia de atraso. **Apenas um integrante do grupo deve ser responsável por enviar o trabalho.**
- O assunto da mensagem deve conter “**Trabalho PBC:nomes dos integrantes do grupo**”.

Importante:

- Trabalhos que não compilam não serão corrigidos.
- Trabalhos evidentemente iguais receberão nota zero.

Critérios para correção: identificação, identificadores adequados para as variáveis, constantes e funções, modularização, interface com o usuário, utilização adequada das bibliotecas organizadas e aplicação dos conceitos de passagens de parâmetros e retorno de função.