Universidade Tecnológica Federal do Paraná — **Disciplina: Estrutura de Dados 1 (EDCO3A)**
Engenharia de Computação — **Professora: Tamara Angélica Baldo**

EXERCÍCIOS DE REVISÃO

Nome: _____ **RA:** _____

Instruções:

- Esboce em um papel a sequência de passos necessários para criar o seu programa. Isso ajuda a programar a solução;
- Crie um arquivo .c para cada um dos exercícios;
- Estes exercícios não estão valendo nota. Sendo assim, não precisam ser postados no Moodle.

(1) [**Recursão**] Escreva e programe uma função recursiva para calcular o valor de um número inteiro de base x elevada a um expoente inteiro y, sendo os valores de $x>0$ e $y>0$ fornecidos pelo usuário.

(2) [**Arquivos**] Faça um programa que receba, por argumento na main, o nome de um arquivo texto. Crie outro arquivo texto de saída contendo o texto do arquivo de entrada original, porém substituindo todas as vogais 'a' pelo caractere '*' e as demais vogais por '+'. Além disso, mostre na tela quantas linhas esse arquivo possui. Dentro do programa faça o controle de erros, isto é, insira comandos que mostre se os arquivos foram abertos com sucesso e, caso contrário, imprima uma mensagem de erro encerrando o programa.

(3) [**Alocação Dinâmica**] Faça um programa que leia um valor N e crie dinamicamente um vetor com essa quantidade de elementos. Em seguida, passe esse vetor para um procedimento que vai preencher os elementos desse vetor com números aleatórios de 0 até N. Depois, no programa principal, imprima os valores do vetor preenchido. Além disso, antes de finalizar o programa, lembre-se de liberar a área de memória alocada.

(4) [**Structs e Ponteiros**] O baralho é frequentemente usado em vários jogos para entretenimento. Este pode ser codificado por meio de dois tipos abstrato de dados:

i) CARTA: que representa uma carta física do baralho. Possui três atributos: símbolo/valor, o naipe, e uma variável booleana indicando se a carta já foi ou não utilizada/jogada;

ii) BARALHO: uma estrutura que representa um conjunto de Cartas.

O trecho de código-fonte abaixo mostra possíveis representações para estas estruturas em linguagem C

```
typedef struct {
    char valor;
    char naipe;
    bool foiJogada;
} Carta;
```

```
typedef struct {
    Carta array[54];
} Baralho;
```

A partir das definições dos tipos pode-se modelar o objeto **Baralho**. Para que um baralho seja manipulado adequadamente ele precisa de funções/procedimentos que mudem seu estado/configuração. Por exemplo, temos que:

- criar o baralho;
- adicionando as cartas;
- consultar a carta do topo;

- consultar a carta do fundo;
- embaralhar novamente as cartas;
- retirar cartas e entregar para os jogadores

Assim sendo, escreva funções/procedimentos em C para simular os comportamentos listados na Tabela 1. Adicione comandos na função principal que testem e validem todas as funções implementadas.

Funções / Procedimentos	**Descrição**
void criaBaralho(baralho *baralho);	inicia um novo baralho criando todas as cartas nele contido.
int cartasNaoJogadas(Baralho *baralho);	Consulta o número de cartas disponíveis para jogo.
Carta topo(Baralho *baralho);	Consulta a carta do topo de um baralho.
Carta fundo(Baralho *baralho);	Consulta a carta do fundo de um baralho.
Carta* carteado(Baralho *baralho);	Retorna um array com 3 cartas aleatórias para um jogador;

Tabela 1: Operações básicas do baralho