

Disciplina de Estrutura de Dados III

Prof. Anderson Canale Garcia

Monitora Beatriz Aimee Teixeira Furtado Braga

Monitor Gustavo Lelli

Monitor Lucas Piovani

Monitor Rafael Freitas Garcia

Primeiro Exercício Prático

Neste exercício, você desenvolverá um sistema para gerenciar e proteger as espécies marinhas em uma reserva natural submarina. O objetivo é monitorar diferentes espécies de animais marinhos, suas localizações, e o impacto de atividades humanas, como pesca e poluição, na saúde do ecossistema.

Cada espécie marinha monitorada possui as seguintes características:

SPECIES_ID (i.e., identificador único da espécie)
NAME (i.e., nome comum da espécie)
SCIENTIFIC_NAME (i.e., nome científico da espécie)
POPULATION (i.e., número de indivíduos na reserva)
STATUS (i.e., estado de conservação: SAUDEL, AMEACADA, CRITICA)
LOCATION (i.e., localização onde a espécie foi avistada, em coordenadas X e Y)
HUMAN_IMPACT (i.e., nível de impacto humano na área: 1, 2 ou 3)

Considere o tipo de arquivo de dados com **campos de tamanho fixo em bytes e registros de tamanho fixo em bytes**. Implemente um programa na linguagem C que ofereça as funcionalidades descritas a seguir:

[1] **REGISTRAR ESPÉCIE MARINHA:** Permita a gravação de novas espécies marinhas no sistema, em formato binário. Não é permitido armazenar todos os registros em memória para escrevê-los em uma única operação, você pode gravar os registros no arquivo a medida em que eles são lidos do teclado. Cada espécie deve ter um SPECIES_ID único. Os campos de STATUS, HUMAN_IMPACT e POPULATION podem ser nulos.

[2] **RELATÓRIO DE ESPÉCIES:** Permita a recuperação dos dados de todas as espécies armazenadas no arquivo de dados, mostrando-os conforme a saída padrão, que garante a distinção dos campos e registros.

[3] **BUSCA DE ESPÉCIE:** Dado o valor de um RRN (número relativo do registro), recupere o registro mostrando seus dados de forma organizada na saída padrão para permitir a distinção dos campos.

[4] **REGISTRAR INFORMAÇÃO:** Implemente uma funcionalidade que permita registrar informações sobre a espécie que poderiam não estar disponíveis no momento de cadastro. O programa deve receber o SPECIES_ID e o dado a ser adicionado. Se o dado já tiver sido cadastrado, deve-se mostrar a mensagem de erro. Caso contrário, escreva no arquivo binário a informação inserida

Entrada:

A entrada do programa consiste em um comando a ser executado (1, 2, 3 ou 4), o nome de um arquivo a ser manipulado e entradas adicionais para cada comando:

Comando 1:

Recebe o número **n** de registros a serem lidos da entrada padrão e escritos no arquivo de saída. Além disso, recebe os **n** registros a serem gravados, cada um com seus campos separados em linhas diferentes.

Exemplo:

1 saída.bin % (Comando 1 / Nome do Arquivo: saída.bin)

1 %(Quantas espécies serão inseridas)

424242 %(Species_ID)

Cachalote %(Nome Comum)

Physeter macrocephalus %(Nome científico)

42 %(número de indivíduos da reserva)

SAUDAVEL %(estado de conservação)

42.0 420.50 %(Coordenadas X e Y)

2 %(Nível de Impacto Humano)

Obs: Campos nulos inteiros são inseridos como o inteiro 0, e campos nulos de string são inseridos como NULO.

Comando 2:

Não recebe entrada adicional

Exemplo:

2 entrada.bin

Comando 3:

Recebe o RRN do registro a ser lido do arquivo.

Exemplo:

3 entrada.bin

0 %(Valor do RRN)

Comando 4:

Recebe a quantidade **n** de informações inseridas e o SPECIES_ID correspondente. Para cada informação inserida, lê-se a instrução “STATUS” “HUMAN IMPACT” ou “POPULATION” e o dado referente.

Exemplo:

4 entrada.bin

40242 %(id da espécie que terá as informações inseridas)

2 %(Quantas informações serão inseridas)

HUMAN IMPACT %(instrução 1)

2 %(dado de impacto humano)

POPULATION %(instrução 2)

40 %(dado de população)

Saídas:

A saída para os comandos 2 e 3 devem ter o seguinte formato para cada registro printado:

ID: “ID da espécie escrito aqui”\n

Nome: “nome comum da espécie escrito aqui”\n

Nome Científico: “nome científico escrito aqui”\n

População: “número de indivíduos escrito aqui”\n

Status: “modo conservação escrito aqui”\n

Localização: “(coord. X, coord. Y)”\n

Impacto Humano: “impacto escrito aqui”\n\n

OBS: As aspas não estão presentes na saída. Campos nulos devem ser impressos como “NULO\n”.

A saída para os comandos 1 e 4 serão feitas utilizando a função `binarioNaTela`, que será fornecida para você utilizar em seu programa. Essa função recebe o nome de um arquivo e printa um double calculado utilizando os bytes presentes no arquivo passado como parâmetro. Use-a após fechar o arquivo escrito.

Saídas de Erro:

Erros como arquivos inexistentes e RRNs inválidos devem apresentar saídas diferentes: para RRNs inválidos ou `SPECIES_ID` inexistentes, a saída deverá ser “Espécie não encontrada”, enquanto que para abertura de arquivos inexistentes a saída deverá ser “Falha no processamento do arquivo”. Para inserção de dados já cadastrados, a mensagem deve ser “ Informação já inserida no arquivo”. Quaisquer outros erros, como entradas do usuário fora do esperado, devem emitir a mensagem de erro padrão “Falha no processamento do arquivo”.

Detalhes:

1. Os tamanhos dos campos do registro de espécies são os seguintes:
 - a. `SPECIES_ID` -> 4 bytes (int)
 - b. `NAME` -> 41 bytes
 - c. `SCIENTIFIC_NAME` -> 61 bytes
 - d. `POPULATION` -> 4 bytes (int)
 - e. `STATUS` -> 9 bytes
 - f. `LOCATION` -> 8 bytes (2 floats de 4 bytes)
 - g. `HUMAN_IMPACT` -> 4 bytes (int)
2. O ‘\0’ das strings deve ser armazenado no arquivo de dados.
3. O valor do float deve ser gravado em duas casas decimais.
4. É necessário preencher o espaço restante do campo do registro com um caractere de lixo, que será o ‘\$’.