

Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
Departamento de Ciências de Computação
Disciplina de Organização de Arquivos
Profa. Dra. Cristina Dutra de Aguiar
Monitor Lucas Romero
Monitor Eduardo Souza
Monitor Voluntário João Paulo Clarindo

Primeiro Exercício Prático

Considere os seguintes dados a respeito de uma pessoa:

FIRSTNAME (i.e., primeiro nome da pessoa)
LASTNAME (i.e., sobrenome da pessoa)
EMAIL (i.e., endereço com e-mail da pessoa)
NATIONALITY (i.e., país onde a pessoa nasceu)
AGE (i.e., idade)

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] Permita a gravação dos dados de várias pessoas em formato binário. Não é necessário 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.

[2] Permita a recuperação dos dados, de todas as pessoas, armazenados no arquivo de dados, mostrando os dados de forma organizada na saída padrão para permitir a distinção dos campos e registros.

[3] 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.

Entrada:

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

Comando 1:

Recebe o número 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 saida.bin

1

Lucas

Romero

lucasromero@usp.br

Brasileiro

20

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

Saídas:

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

Firstname: "primeiro nome do registro escrito aqui"\n

Lastname: "último nome do registro escrito aqui"\n

Email: "email do registro escrito aqui"\n

Nationality: "nacionalidade do registro escrito aqui"\n

Age: "idade do registro escrito aqui"\n\n

OBS: as aspas não estão presentes na saída.

A saída para o comando 1 será feita 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.

Saídas de Erro:

Erros como arquivos inexistentes e RRNs inválidos devem apresentar saídas diferentes: para RRNs inválidos, a saída deverá ser "Não foi possível ler o arquivo", enquanto que para arquivos inexistentes a saída deverá ser "Falha no processamento do arquivo".

Detalhes:

1. Os tamanhos dos campos do registro de pessoa são os seguintes:
 - a. Firstname -> 51 bytes
 - b. Lastname -> 51 bytes
 - c. Email -> 81 bytes
 - d. Nationality -> 51 bytes
 - e. Age -> 4 bytes (int)
2. O '\0' das strings deve ser armazenado no arquivo de dados.
3. É necessário preencher o espaço restante do campo do registro com um caractere de lixo, que será o '\$'.