

Exercício 2 - Structs

- 1) Nesse exercício vamos trabalhar, de forma simplificada, com número complexos. Números complexos são aqueles que são escritos na forma $a + bi$, onde a e b são números reais, e i denota uma unidade imaginária. a é a parte real do número complexo, enquanto b é conhecido como a parte imaginária.
- Defina uma estrutura para representar números complexos. (Dica: sua estrutura precisa somente armazenar a parte real e a parte imaginária do número complexo.)
 - Escreva um procedimento de nome `printComplexo()`, que receba como parâmetro um número complexo (a estrutura definida previamente) e imprima esse número. A impressão deve ser feita no seguinte formato:
$$(a + bi)$$
onde a e b são os números do número complexo.
 - Escreva uma função de nome `somaComplexo()`, que receba como parâmetro dois números complexos e retorne um terceiro número complexo, que representa a soma dos números passados como parâmetro. A soma de números complexos segue a seguinte forma:
$$(a + bi) + (c + di) = (a + c) + (b + d)i$$
ou seja, por exemplo, $(3 + 4i) + (2 - i) = 5 + 3i$.
 - Usando as funções e a estrutura previamente definidas, escreva um programa que leia dois números complexos via `scanf()` e imprima a soma desses dois números. Segue um exemplo de entrada e saída.
Forneca a parte real e imaginaria de um numero complexo: 20 5
Forneca a parte real e imaginaria de um numero complexo: 3 2
 $(20 + 5i) + (3 + 2i) = (23 + 7i)$
- 2) Nesse exercício você deve escrever um programa em C que irá trabalhar com diretores de filmes. Um diretor deve possuir informações sobre seu nome (máximo de 20 letras), a quantidade de filmes dirigidos, e cada um dos filmes dirigidos por ele (considere que um diretor tem, no máximo, 100 filmes). Um filme deve possuir informações sobre seu nome (máximo de 50 caracteres), sua duração em minutos e sua data de lançamento. Uma data é formada pelo número do dia, do mês e do ano.
- Crie em seu programa as estruturas necessárias para armazenar as informações sobre filmes e diretores.
 - Declare em seu programa um vetor de 3 posições de diretores. Em seguida, inicialize esse vetor com dados fornecidos pelo usuário.

- c. Crie um procedimento de nome `imprimeFilme()`, que imprime todos os dados relativos a certo filme passado como parâmetro. Tenta deixar a saída do seu procedimento apresentável!
 - d. Crie um procedimento de nome `imprimeDiretor()`, que imprime todos os dados relativos a certo diretor passado como parâmetro (incluindo a informação de cada um dos filmes produzidos por ele). Tenta deixar a saída do seu procedimento apresentável!
 - e. Usando os procedimentos previamente declarados, imprima as informações de todos os diretos declarados em b).
- 3) Nesse exercício iremos escrever um programa para trabalhar com bandas de música.
- a. Defina uma estrutura de nome `Banda`, que irá representar bandas de música. Essa estrutura deve ter como campos o nome da banda (máximo 20 caracteres), que tipo de música ela toca (máximo 20 caracteres), o número de integrantes dessa banda e um número inteiro que representa o ranking dessa banda.
 - b. Defina, no método `main()` de seu programa, um vetor de 5 posições do tipo `Banda`, e inicialize as 5 posições desse vetor. Utilize a diretiva `define` para definir o tamanho do vetor declarado.
 - c. Crie um procedimento de nome `imprimeBanda()`, que recebe como parâmetro uma variável do tipo `Banda` e imprima todas as informações relativas a essa banda.
 - d. Crie uma função de nome `bandaPorRanking()`, que receba como parâmetros um vetor de 5 posições da estrutura `Banda` e um número inteiro, que representa um ranking. Sua função deve procurar no vetor uma `Banda` com o ranking passado como parâmetro, e retornar essa `Banda`. Considere que o vetor passado como parâmetro nessa função tem sempre 5 posições. Teste seu programa usando essa função e imprimindo todas as informações da banda retornada com a função implementada em c).
 - e. Crie uma função de nome `buscaBanda()`, que receba como parâmetros um vetor de 5 posições da estrutura `Banda` e uma string com o nome de uma banda. Sua função deve procurar no vetor de `Bandas` por uma banda com o nome passado como parâmetro, e retornar a `Banda` em questão. Considere que o vetor do tipo `Banda` passado como parâmetro nessa função tem sempre 5 posições. Teste seu programa usando essa função e imprimindo todas as informações da banda retornada.