



Aula Prática P-09

- * Todos os exercícios que envolvem programas devem ser resolvidos através de programas em C/C++.
 - * A entrega será feita até às 23h55 do dia da aula prática através do Moodle, sem zipar (entregue apenas o código fonte).
 - * Inclua seu número de matrícula, nome e turma em um comentário no início de cada arquivo com código fonte.
 - * Você só pode utilizar conhecimento prévios à aula para resolver o exercício. Caso use uma matéria que ainda não foi dada sua nota será penalizada.
 - * Códigos que não compilam serão zerados.
-

Questão 01

Implemente uma função `equal` que retorna o inteiro 1 se dois números racionais, r_1 e r_2 , são iguais e 0 caso contrário. Dica: reduza r_1 e r_2 a seus termos mínimos (lembra do MDC?) e verifique em seguida se os termos são iguais.

Implemente o método `main` para ler e comparar os números racionais r_1 e r_2 , representados pela estrutura a seguir:

```
1 struct Racional {  
2     int numerador;  
3     int denominador;  
4 };
```

Exemplos de execução:

```
1 Digite numerador e denominador de r1: 100 40  
2 Digite numerador e denominador de r2: 5 2  
3  
4 r1 e r2 são iguais!
```

```
1 Digite numerador e denominador de r1: 100 40  
2 Digite numerador e denominador de r2: 5 3  
3  
4 r1 e r2 são diferentes!
```

Questão 02

Os números complexos são escritos na sua forma algébrica como $a + bi$, em que $i = \sqrt{-1}$. Assim, a e b são números reais, sendo a a parte real do número complexo e b a parte imaginária. Portanto, um número complexo z pode ser escrito como $z = a + bi$.

Implemente em C as funções a seguir seguindo os protótipos abaixo. Em seguida, implemente o método `main`, responsável por solicitar ao usuário dois números complexos e perguntar qual operação deve ser feita (soma, subtração ou multiplicação).

```
1 typedef struct {
2     double real;
3     double imaginario;
4 } Complexo;
5
6 // funcao que retorna x + y
7 Complexo somaComplexo(Complexo x, Complexo y);
8
9 // funcao que retorna x - y
10 Complexo subComplexo(Complexo x, Complexo y);
11
12 // função que retorna x * y
13 Complexo multComplexo(Complexo x, Complexo y);
```

Dica: lembre-se que para efetuar a soma de dois números complexos deve-se somar a parte real e a parte imaginária de cada número separadamente. No caso da multiplicação, lembre-se que:

$(a + bi) \times (c + di) = ac + adi + bci + bdi^2 = ac + adi + bci - bd = ac - bd + (ad + bc)i$ (lembrando que $i^2 = -1$).

Exemplos de execução:

```
1 Calculadora de números complexos!
2
3 Digite os valores de a e b (x = a + bi): 2 5
4 Digite os valores de c e d (y = c + di): 1 -2
5 Digite a operação (+, - ou *): *
6
7 Resultado: 12 + i
```

```
1 Calculadora de números complexos!
2
3 Digite os valores de a e b (x = a + bi): 10 10
4 Digite os valores de c e d (y = c + di): 10 -10
5 Digite a operação (+, - ou *): *
6
7 Resultado: 200
```

Questão 03

Crie um novo tipo chamado **Aluno** para representar um aluno e pode escolher o formato do arquivo texto gerado, desde que toda a informação seja devidamente armazenada. Em seguida, leia o **nome**, **nota** e **frequência** de n alunos e armazene estes dados em um arquivo texto.

Questão 04

Usando o arquivo do exercício anterior, faça um com as seguintes funcionalidade que leia o arquivo criado e imprime:

1. A nota média dos alunos.
2. A frequência média dos alunos.
3. O percentual de alunos aprovados.
4. Os nomes dos alunos com nota acima da nota média (calculada no item 1).

Exemplo de execução (dados digitados pelo usuário estão destacados em azul):

```
1 Digite o nome do arquivo: alunos.txt
2
3 Nota média: 7.2
4 Frequência média: 82%
5 Percentual de aprovação: 70%
6
7 Nomes dos alunos com nota acima da nota média:
8 Aluno_1
9 Aluno_2
10 Aluno_5
11 Aluno_6
12 Aluno_9
```

Questão 05

Crie um programa que lê o arquivo texto criado no exercício 3 e converte para binário, ou seja, cria um novo arquivo binário e insere os mesmos dados do arquivo original.

Em seguida, o programa calcula e imprime:

1. A nota média dos alunos.
2. A frequência média dos alunos.
3. O percentual de alunos aprovados.
4. Os nomes dos alunos com nota acima da nota média (calculada no item 1).

Exemplo de execução (dados digitados pelo usuário estão destacados em azul):

```
1 Digite o nome do arquivo texto: alunos.txt
2 Arquivo alunos.dat criado com sucesso!
3
4 Nota média: 7.2
5 Frequência média: 82%
6 Percentual de aprovação: 70%
7
8 Nomes dos alunos com nota acima da nota média:
9 Aluno1
10 Aluno2
11 Aluno5
12 Aluno6
13 Aluno9
```