



Lista 1 - Funções

Exercícios para sala de aula

Observação: Todos os programas devem conter a opção de repetição de programa.

1) Fazer um programa que permite aplicar as operações de soma, subtração, multiplicação, divisão e resto em dois números fornecidos pelo usuário. Cada operação é uma função e deve ser acessada a partir de um menu (use *switch case*). As funções são implementadas no próprio programa.

- a) Soma sem parâmetros e sem retorno
- b) Subtração com parâmetros e sem retorno
- c) Multiplicação sem parâmetros e com retorno
- d) Divisão com parâmetros e com retorno.
- e) Resto com parâmetros e com retorno.

Exemplo de entrada e saída:

```
A - Adicionar  
S - Subtrair  
M - Multiplicar  
D - Dividir  
R - Resto
```

```
Opcao: a
```

```
Informe o primeiro valor: 2
```

```
Informe o segundo valor: 3
```

```
Soma: 5
```

```
Executar novamente ($ ou N): s
```

```
A - Adicionar  
S - Subtrair  
M - Multiplicar  
D - Dividir  
R - Resto
```

```
Opcao: s
```

```
Informe um numero: 10
```

```
Informe outro numero: 7
```

```
Subtracao: 3
```

```
Executar novamente (S ou N): s
A - Adicionar
S - Subtrair
M - Multiplicar
D - Dividir
R - Resto
Opcao: m
Informe o primeiro valor: 6
Informe o segundo valor: 3
Multiplicacao: 18
```

```
Executar novamente (S ou N): s
A - Adicionar
S - Subtrair
M - Multiplicar
D - Dividir
R - Resto
Opcao: d
Informe um numero: 12
Informe outro numero: 4
Divisao: 3.00
```

```
Executar novamente (S ou N): s
A - Adicionar
S - Subtrair
M - Multiplicar
D - Dividir
R - Resto
Opcao: r
Informe um numero: 12
Informe outro numero: 2
Resto: 0
```

```
Executar novamente (S ou N):
```

2) Fazer uma função para verificar se um número é ou não primo. A função será implementada em um arquivo de cabeçalho "primo.h" e deverá retornar 0 se o número for primo ou 1, caso o número não seja primo. Elaborar um programa para usar essa função para:

- Verificar se um número informado pelo usuário é ou não um número primo. Validar a entrada para que o usuário informe um número positivo.
- Mostrar os primos no intervalo entre 1 e 100.
- Fazer a média dos primos entre 200 e 100.

Exemplo de entrada e saída:

```
A - Verificar se um numero eh primo
B - Primos entre 1 e 100
C - Media dos primos entre 200 e 100
Opcao: a
```

```
Informe um numero: 10
10 nao eh primo
```

```
Executar novamente (S ou N): s
A - Verificar se um numero eh primo
B - Primos entre 1 e 100
C - Media dos primos entre 200 e 100
Opcao: b
2      3      5      7      11     13     17     19     23     29
31     37     41     43     47     53     59     61     67     71
73     79     83     89     97
```

```
Executar novamente (S ou N): s
A - Verificar se um numero eh primo
B - Primos entre 1 e 100
C - Media dos primos entre 200 e 100
Opcao: c
```

```
Media dos primos entre 200 e 100: 150.81
```

```
Executar novamente (S ou N): n
```

3) Fazer duas funções: uma para calcular o fatorial de um número e a outra para mostrar o fatorial de um número. Ambas as funções devem ser implementadas em um arquivo de cabeçalho chamado "fatorial.h".

Elaborar um programa que use essas funções para:

- Calcular e mostrar o fatorial dos números entre 1 e 8.
- Calcular e mostrar o fatorial de um valor informado pelo usuário. Continuar a leitura enquanto forem informados valores positivos.

Exemplo de entrada e saída:

```
A - Fatorial entre 1 e 8
B - Fatorial de um numero
Opcao: a
```

```
1! = 1 = 1
2! = 2 * 1 = 2
3! = 3 * 2 * 1 = 6
4! = 4 * 3 * 2 * 1 = 24
5! = 5 * 4 * 3 * 2 * 1 = 120
6! = 6 * 5 * 4 * 3 * 2 * 1 = 720
7! = 7 * 6 * 5 * 4 * 3 * 2 * 1 = 5040
8! = 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1 = 40320
```

```
Executar novamente (S ou N): s
A - Fatorial entre 1 e 8
B - Fatorial de um numero
Opcao: b
```

```
Informe um numero: 10
10! = 10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1 = 3628800
```

```
Informe um numero: 12
12! = 12 * 11 * 10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1 = 479001600
```

```
Informe um numero: 0
```