

## Exercícios 01 :: Básico da Linguagem C

### Instruções Gerais

- Os exercícios são de resolução individual.
  - Cria uma pasta para a lista e faça cada exercício em um arquivo distinto.
  - Utilize a extensão .c e o compilador gcc. Utilize o editor VS Code ou outro de sua preferência.
  - Não é permitido o uso de recursos que ainda não foram abordados na disciplina até o momento da publicação desta lista. Esta lista considera: variáveis e tipos, comandos de seleção e funções.**
- Escreva um programa que lê (pelo terminal) dois números inteiros (a e b) e informa:
    - Adição
    - Subtração (a menos b)
  - Escreva um programa que lê dois números inteiros (a e b) e informa:
    - Multiplicação
    - Divisão inteira (a dividido por b)
    - Divisão float (a dividido por b)
  - Escreva um programa que lê o raio (r) de um círculo. O programa deve informar: o diâmetro (2r), a circunferência ( $2\pi r$ ) e a área do círculo ( $\pi r^2$ ). Crie uma macro: `#define PI 3.141593`
  - Escreva um programa que lê dois números inteiros (a e b) e informa:
    - Resto (utilizado o operador %)
    - Resto (sem utilizar o operador %)
      - Dica: Faça a divisão “no papel” e observe quais outras operações podem ser utilizadas para obter o resto.
  - Escreva uma função que imprime a tabuada de um número informado. Dica: lembre-se que você pode alinhar os valor na saída com `%2d` (dois dígitos, alinhados à direita)

**Função:** `void print_mult_table(int number)`

Ex: `print_mult_table(5)`

5 x 1 = 5	5 x 6 = 30
5 x 2 = 10	5 x 7 = 35
5 x 3 = 15	5 x 8 = 40
5 x 4 = 20	5 x 9 = 45
5 x 5 = 25	5 x 10 = 50

- Escreva uma função que recebe um float e informa, separadamente:
  - O número com 2 casas de precisão;
  - A parte inteira e a parte decimal;
  - Sua representação em notação científica (mantissa/expoente)\*\*

\*\*Verifique a documentação da função printf: <http://www.cplusplus.com/reference/cstdio/printf/>

**Função:** `void print_float(float number)`

7. Escreva uma função que recebe um número de dias e informa a quantidade correspondente em: anos + semanas + dias.
- Considere:
    - Ano = 365 dias
    - Semana = 7 dias
  - Exemplo:  
Dias: 427 = 1 ano(s), 8 semana(s) e 6 dia(s)

**Função:** `void print_days(int days)`

8. Escreva uma função que recebe um inteiro representando um ano e retorna se o mesmo é bissexto. A função deve retornar 1 (true) para bissexto ou 0 (false), em caso contrário. Para um ano ser bissexto (leap year), ele precisa:
- Ser divisível por 4 e não divisível por 100, OU, ser divisível por 400.

**Função:** `int is_leap_year(int year)`

9. Escreva uma função que recebe três números e os imprime em ordem crescente.

**Função:** `void print_sorted(int a, int b, int c)`

10. Escreva uma função que recebe três notas escolares n1, n2 e n3 (0..10). A função deve calcular e imprimir a média aritmética simples das três notas, bem como, o conceito que o aluno obteve pela média, segundo estes critérios:
- Conceito A, se média no intervalo [8,5..10]
  - Conceito B, se média no intervalo [7,0..8,5[
  - Conceito C, se média no intervalo [5,5..7,0[
  - Conceito F, se média inferior a 5,5

**Função:** `void print_grade(float n1, float n2, float n3)`

11. Escreva um programa que lê um caractere da entrada. Se não for letra, informa. Se for letra, verifica se é maiúscula (e passa para maiúscula se necessário). Ao final, deve informar a letra digitada em maiúsculo.
12. Escreva um programa que lê um caractere e informa:
- Se é letra e, neste caso, também informa se é vogal ou consoante;
  - Se é número;
  - Se é símbolo (ASCII 33 ao 126, exceto números e símbolos).
13. Escreva um programa que lê um inteiro de até 4 dígitos (positivo ou negativo) e o imprime por extenso. OBS: para extrair os dígitos do número, utilize **resto por 10** (para obter a unidade) e **divisão por 10** (para remover a unidade).
14. Escreva um programa que calcula a idade de uma pessoa a partir da data de nascimento, lida pelo terminal em três variáveis (dia, mês e ano). Para obter a data atual, utilize as funções **get\_day()**, **get\_month()** e **get\_year()**, fornecidas pelo código em anexo.

```
#include<time.h>

struct tm get_date() {
    time_t t = time(NULL);
    return *localtime(&t);
}

int get_day() {
    return get_date().tm_mday;
}

int get_month() {
    return get_date().tm_mon+1;
}

int get_year() {
    return get_date().tm_year+1900;
}
```