

1. Tia Maria estava servindo suco para as crianças da sua sala no jardim de infância quando notou que tinha colocado no copo do Joãozinho suco de abacaxi e no copo da Mariazinha suco de melancia. Ocorre que Tia Maria se confundiu: Joãozinho não gosta de suco de abacaxi, e Mariazinha não toma suco de melancia. Como as jarras de suco já estão vazias – na verdade, já foram até lavadas – Tia Maria precisa trocar o conteúdo dos copos, já que cada criança só deve beber do seu próprio copo. Descreva, usando linguagem natural, um algoritmo para que Tia Maria realize a troca do conteúdo dos dois copos. Suponha que os dois copos têm o mesmo tamanho, e que Tia Maria deve sujar o mínimo possível de louça.

2. Descreva, usando linguagem natural e notação matemática, um algoritmo para decompor um dado número de segundos em dias, horas, minutos e segundos. Por exemplo, 90061 segundos equivalem a 1 dia, 1 hora, 1 minuto e 1 segundo. Já 86401 segundos equivalem a 1 dia, 0 hora, 0 minuto e 1 segundo. A repostagem deve ser aplicável a qualquer quantidade de segundos!

3. Instale a IDE Code::Blocks no seu computador. Para instalação em sistemas Windows, faça o download no site:

<https://www.codeblocks.org/downloads/binaries/>

Certifique-se de instalar uma versão que inclui o pacote mingw. Se você não tem privilégios de administrador, instale uma versão nonadmin. Também observe que o caminho da instalação **não deve** conter acentos ou o caractere ç.

Caso você tenha experiência e prefira utilizar outra ferramenta (exemplo: Visual Studio), você pode usá-la, desde que a) saiba como criar programas a partir de múltiplos arquivos-fonte, de preferência por meio de um Makefile; e b) saiba como usar um debugger associado à ferramenta. Em sistemas Linux, você pode utilizar o compilador gcc e o debugger gdb. Entretanto, note que os professores não darão suporte caso você resolva utilizar uma IDE que não seja o Code::Blocks.

Uma vez tendo preparado um ambiente para desenvolvimento, implemente, compile e execute o programa que mostra a mensagem `Hello World!` na tela - sim, seu primeiro programa deve ter exatamente esta mensagem: tradição é tradição!

Depois, repita a tarefa utilizando um editor online (exemplo: <https://replit.com>).

4. Dado o código-fonte do "Hello World!":

```
#include <stdio.h>

int main ()
{
    printf ("Hello world!\n");

    return 0;
}
```

a) O que acontece se tirarmos o `\n` do argumento do `printf`?

b) O que acontece se em vez do `\n` tentarmos simplesmente pular uma linha, como abaixo?

```
printf ("Hello world!
");
```

c) O que acontece se colocarmos um `\n` entre as palavras, como abaixo?

```
printf ("Hello\nworld!\n");
```

d) O que acontece se usarmos 2 `printf`'s?

```
printf ("Hello ");
printf ("world!\n");
```

e) Ajuste o programa para testar todas as chamadas da função `printf` abaixo. Depois, indique quais opções imprimem na tela **exatamente** a mesma saída (ou seja, mesma mensagem com a mesma formatação) que:

```
printf ("Hello\n world!\n");
```

I) `printf("Hello\t world!\n");`

II) `printf ("Hello"); printf(" world!\n");`

III) `printf ("Hello\n world!\n");`

IV) `printf ("Hello\n world!\n");`

V) `printf("Hello\n"); printf(" world!\n");`

5. O programa abaixo deveria calcular o IMC (definido como o peso dividido pelo quadrado da altura). Entretanto, o resultado apresentado é incorreto. Por quê?

```
#include <stdio.h>

int main ()
{
    printf ("Calculo do IMC \n");
    printf ("Altura: %f \n", 1.64);
    printf ("Peso: %f \n", 62.5);

    printf ("IMC: \n", 62.5 / 1.64 * 1.64);

    return 0;
}
```

6. O código abaixo deveria trocar o conteúdo de duas variáveis, mas ambas terminam com o mesmo valor! O que está errado? Como resolver o problema?

```
1.  #include <stdio.h>
2.
3.  int main ()
4.  {
5.      int primeiro = 10,
6.          segundo = 20;
7.
8.      primeiro = segundo;
9.      segundo = primeiro;
10.
11.     return 0;
12. }
```

7. Desafio! Um agricultor possui 1 (uma) espiga de milho. Cada espiga tem 150 grãos, e cada grão pesa 1g (um grama). Descreva, usando linguagem natural, um algoritmo para determinar quantos anos serão necessários para que o agricultor colha mais de cem toneladas de milho (1T = 1000Kg, 1Kg = 1000g), sendo que: (a) A cada ano ele planta todos os grãos da colheita anterior. (b) Há apenas uma colheita por ano. (c) 10% (dez por cento) dos grãos não germinam (morrem sem produzir). (d) Cada grão que germina produz duas espigas de milho. Considere que a quantidade de terra disponível é sempre suficiente para o plantio.