

1. Tia Maria vai precisar de um recipiente auxiliar para colocar o conteúdo de um dos copos. Pode ser outro copo que tenha pelo menos o mesmo tamanho dos copos das crianças, ou até uma das jarras. Basta então ela colocar o conteúdo do copo de Mariazinha (suco de melancia) no recipiente auxiliar, o conteúdo do copo do Joãozinho (suco de abacaxi) no copo de Mariazinha, e o conteúdo que estava no recipiente auxiliar (suco de melancia) no copo de Joãozinho. Se precisar, ela pode lavar os copos entre as operações. O conteúdo dos copos fica assim:

Início:	Joãozinho: abacaxi	Mariazinha: melancia	Auxiliar: nada
1 troca:	Joãozinho: abacaxi	Mariazinha: nada	Auxiliar: melancia
2 trocas	Joãozinho: nada	Mariazinha: abacaxi	Auxiliar: melancia
3 trocas	Joãozinho: melancia	Mariazinha: abacaxi	Auxiliar: nada

2. Dado T o número total de segundos:

- O número D de dias é dado pela parte inteira da divisão $T/86400$ (86400 é o número de segundos em um dia: $60 \times 60 \times 24$).

- O número H de horas é dado pela parte inteira da divisão $(T-86400D)/3600$ (3600 é o número de segundos em uma hora: 60×60).

- O número M de minutos é dado pela parte inteira da divisão $(T-86400D-3600H)/60$.

- O número S de segundos é dado por $T-86400D-3600H-60M$.

3. Não há correção, mas procure ajuda imediatamente caso não tenha conseguido fazer tudo com sucesso!

4.

a) A diferença é sutil - com o `\n`, o programa pula uma linha entre o Hello world! impresso e as informações de execução (Process returned 0...).

b) O programa nem sequer compila. Você não pode "quebrar" a string desta forma, e pular uma linha entre aspas não é igual a fazer o programa pular uma linha.

c) A saída fica sendo:

```
Hello
world!
```

d) A saída fica igual à original. Usar 2 printf's não faz com que uma linha seja pulada, apenas o `\n` tem este efeito.

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

e) As alternativas III e V.

5. Os operadores / e * têm a mesma precedência. Como não usamos parênteses, eles são avaliados da esquerda para a direita. Com isso, o cálculo realizado é equivalente a (peso / altura) * altura, e não peso / (altura*altura), como esperado.

6.

- O comando da linha 5 declarou e inicializou (com o valor 10) a variável chamada primeiro.

- O comando da linha 6 declarou e inicializou (com o valor 20) a variável chamada segundo. Neste ponto, primeiro vale 10 e segundo vale 20.

- O comando da linha 8 substituiu o conteúdo inicial (10) por uma cópia do conteúdo da variável segundo (20). Neste ponto, ambas as variáveis valem 20.

- O comando da linha 9 atribui o conteúdo da variável primeiro à variável segundo. Nada muda, pois as duas variáveis já estão com o valor igual.

Para fazer a troca, precisamos de uma variável auxiliar! Lembra dos copos de suco?

```
#include <stdio.h>
```

```
int main ()
```

```
{
```

```
    int primeiro = 10,  
        segundo = 20,  
        aux;
```

```
    aux = primeiro;  
    primeiro = segundo;  
    segundo = aux;
```

```
    return 0;
```

```
}
```

7. O truque para entender este tipo de exercício é sempre começar listando o estado dos elementos que fazem parte do problema, montando uma tabela. Para identificar quais são estes elementos, precisamos interpretar o enunciado. Fazendo as contas para cada ano, temos o seguinte resultado:

Anos passados	Número de espigas	Número de grãos	Toneladas
0	1	150	$(150/1000)/1000 = 0,00015$
1	$150 \times 0,9 \times 2 = 270$	$270 \times 150 = 40500$	$40500/1000000 = 0,0405$
2	72900	10935000	10,935
3	19683000	2952450000	2952,45

Podemos então identificar como a tabela foi preenchida:

- Inicialmente, temos:

anos passados: 0

espigas de milho: 1

grãos: 150 x o número de espigas de milho

peso (em toneladas): número de grãos / 1000000

- Repita até que o peso seja maior que 100 toneladas:

Aumente em 1 o número de anos passados.

Defina o novo número de espigas como sendo o número de grãos anterior x 0,9 (porque apenas 90% das espigas germina) x 2 (porque cada grão gera 2 espigas).

Defina o novo número de grãos como sendo o novo número de espigas x 150.

Defina o novo peso como sendo o novo número de grãos / 1000000.

É possível eliminar algumas dessas entidades, porque elas são definidas em função de outras (por exemplo, em vez do peso, poderíamos usar diretamente o número de grãos / 1000000), mas por clareza por enquanto fica tudo explícito.