

nível —

# EASY



Arrays

## 7. Organizando Resultados

O sistema interno da nave realiza periodicamente uma verificação de seus principais componentes para checar sua integridade.

Os vários resultados dessa verificação são emitidos em formato de listas de referências numéricas desorganizadas.

Para otimizar isso você deve criar um procedimento que seja capaz de organizar todas as listas de números em uma única lista ordenada.

### Desafio 7: Organizando Resultados

Escreva uma função que recebe um array bidimensional de inteiros e retorna um único array contendo todos os números em ordem ascendente.

## Testes 7: Organizando Resultados

- Entrada: ([ [1, 5, 3], [6, 19, 11], [47, 128, 5], [1, 93, 57, 42, 103] ])  
Saída: [ 1, 1, 3, 5, 5, 6, 11, 19, 42, 47, 57, 93, 103, 128 ]
- Entrada: ([ [1, 3], [4, 8], [7, 5], [2, 6] ])  
Saída: [ 1, 2, 3, 4, 5, 6, 7, 8 ]
- Entrada: [ [], [], [], [] ]  
Saída: [ ]
- Entrada: ([ [100, 50], [60, 100], [20, 100, 70], [10, 40, 80, 90] ])  
Saída: [ 10, 20, 40, 50, 60, 70, 80, 90, 100, 100, 100 ]

Strings

## 8. Biblioteca Interplanetária

A Biblioteca Interplanetária acumula conhecimento de todas as partes da galáxia, possuindo um acervo gigantesco e muito rico.

Todas as publicações são organizadas por autor e para isso o último nome do autor é utilizado para identificação.

A direção da biblioteca precisa que você crie um programa para converter o nome completo de um autor para um nome abreviado corretamente no formato utilizado na organização das publicações.

### Desafio 8: Biblioteca Interplanetária

Escreva uma função que recebe uma string contendo um nome completo e retorna uma string com todos os nomes, exceto o último, abreviados e o último nome em letras maiúsculas antes das abreviações separado por vírgula.

## Testes 8: Biblioteca Interplanetária

- Entrada: ('Isaac Larrubia Ferreira Pontes')

Saída: 'PONTES, I. L. F.'

- Entrada: ('John Ronald Reuel Tolkien')

Saída: 'TOLKIEN, J. R. R.'

- Entrada: ('christopher james paolini')

Saída: 'PAOLINI, C. J.'

- Entrada: ('Suzanne Marie Collins')

Saída: 'COLLINS, S. M.'

Strings

## 9. Instruções de Emergência

As estações e naves de maior porte da Frota Estelar em situações de emergência precisam enviar instruções de navegação importantes e que não podem ser comprometidas.

Para isso, algumas validações precisam ser feitas nos códigos de transmissão.

Uma dessas validações envolve um código que deve possuir a mesma quantidade de cada caractere que o compõe. Foi solicitado a você que crie a função que validará este código.

### Desafio 9: Instruções de Emergência

Escreva uma função que recebe uma string, verifica se ela possui a mesma quantidade de cada letra que a compõe e retorna true caso possua ou false caso não possua.

## Testes 9: Instruções de Emergência

- Entrada: ('This is Thee')

Saída: true

- Entrada: ('ssd')

Saída: false

- Entrada: ('Lorem ipsum')

Saída: false

- Entrada: ('QQwweerrttyy')

Saída: true

Recursão Matemática

## 10. Procedimento Recursivo II

Mais uma vez a equipe de manutenção solicitou a sua ajuda para analisar o desempenho dos computadores portáteis da equipe de reconhecimento.

Dessa vez, no entanto, o procedimento que você ficou encarregado de testar envolve cálculos matemáticos, e você irá utilizar o cálculo do fatorial de um determinado número para isso.

Portanto, será preciso criar um procedimento que seja capaz de calcular o fatorial de qualquer número inteiro positivo sem utilizar estruturas de repetição.

### Desafio 10: Procedimento Recrusivo II

Escreva uma função que recebe um número e retorna o seu fatorial sem utilizar nenhuma estrutura de repetição (while, dowhile, for, etc).

O fatorial de um número é igual a multiplicação de todos os inteiros positivos menores ou iguais a ele. Ela deve ser capaz de retornar números inteiros corretos mesmo para valores altos.

## Testes 10: Procedimento Recursivo II

- Entrada: (5)

Saída: 120n

- Entrada: (0)

Saída: 1n

- Entrada: (32)

Saída: 263130836933693530167218012160000000n

- Entrada: (9n)

Saída: 362880n

Arrays

## 11. Intervalo de Coordenadas

O capitão da sua nave precisa que você implemente no sistema principal uma forma de obter todas as coordenadas abaixo de um determinado ponto partindo da origem (0, 0) e retorne isso para o sistema em uma lista ordenada crescente. As coordenadas são escritas na forma de pares ordenados (x, y).

### Desafio 11: Intervalo de Coordenadas

Escreva uma função que receba um par ordenado (x, y) e retorne um array de pares (x, y) onde cada um deles possui x e y menor ou igual ao par recebido em ordem crescente.

Os pares devem ser ordenados de forma que primeiro aumente o valor de y e depois o valor de x.

Apenas o quadrante onde x e y são positivos deve ser considerado.

## Testes 11: Intervalo de Coordenadas

- Entrada: ([2, 2])

Saída: [[0, 0], [0, 1], [0, 2], [1, 0], [1, 1], [1, 2], [2, 0], [2, 1], [2, 2]]

- Entrada: ([2, 7])

Saída: [[0, 0], [0, 1], [0, 2], [0, 3], [0, 4], [0, 5], [0, 6], [0, 7], [1, 0], [1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [1, 6], [1, 7], [2, 0], [2, 1], [2, 2], [2, 3], [2, 4], [2, 5], [2, 6], [2, 7]]

- Entrada: ([-3, -3])

Saída: []

- Entrada: ([7, 6])

Saída: [[0, 0], [0, 1], [0, 2], [0, 3], [0, 4], [0, 5], [0, 6], [1, 0], [1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [1, 6], [2, 0], [2, 1], [2, 2], [2, 3], [2, 4], [2, 5], [2, 6], [3, 0], [3, 1], [3, 2], [3, 3], [3, 4], [3, 5], [3, 6], [4, 0], [4, 1], [4, 2], [4, 3], [4, 4], [4, 5], [4, 6], [5, 0], [5, 1], [5, 2], [5, 3], [5, 4], [5, 5], [5, 6], [6, 0], [6, 1], [6, 2], [6, 3], [6, 4], [6, 5], [6, 6], [7, 0], [7, 1], [7, 2], [7, 3], [7, 4], [7, 5], [7, 6]]

Matemática

## 12. Espaçoponto Comercial

A sua nave está de passagem por um planeta que usa como dinheiro as moedas de créditos.

Existem moedas de 1, 5, 10, 25, 100 e 500 créditos.

Para conseguir um favor de um comerciante local você irá ajudá-lo a automatizar o processo de contagem das moedas através de um programa que calcula quantas de cada moeda ele irá precisar para chegar a um valor determinado.

### Desafio 12: Espaçoponto Comercial

Escreva uma função que receba um valor inteiro e retorne a quantidade de cada moeda para se chegar ao valor.

Deve-se sempre priorizar as moedas de maior valor (o máximo possível de moedas de 500, depois o máximo possível de moedas de 100, etc).

## Testes 12: Espaçoponto Comercial

- Entrada: (478)

Saída: { '1': 3, '5': 0, '10': 0, '25': 3, '100': 4, '500': 0 }

- Entrada: (384)

Saída: { '1': 4, '5': 1, '10': 0, '25': 3, '100': 3, '500': 0 }

- Entrada: (5412)

Saída: { '1': 2, '5': 0, '10': 1, '25': 0, '100': 4, '500': 10 }

- Entrada: (50)

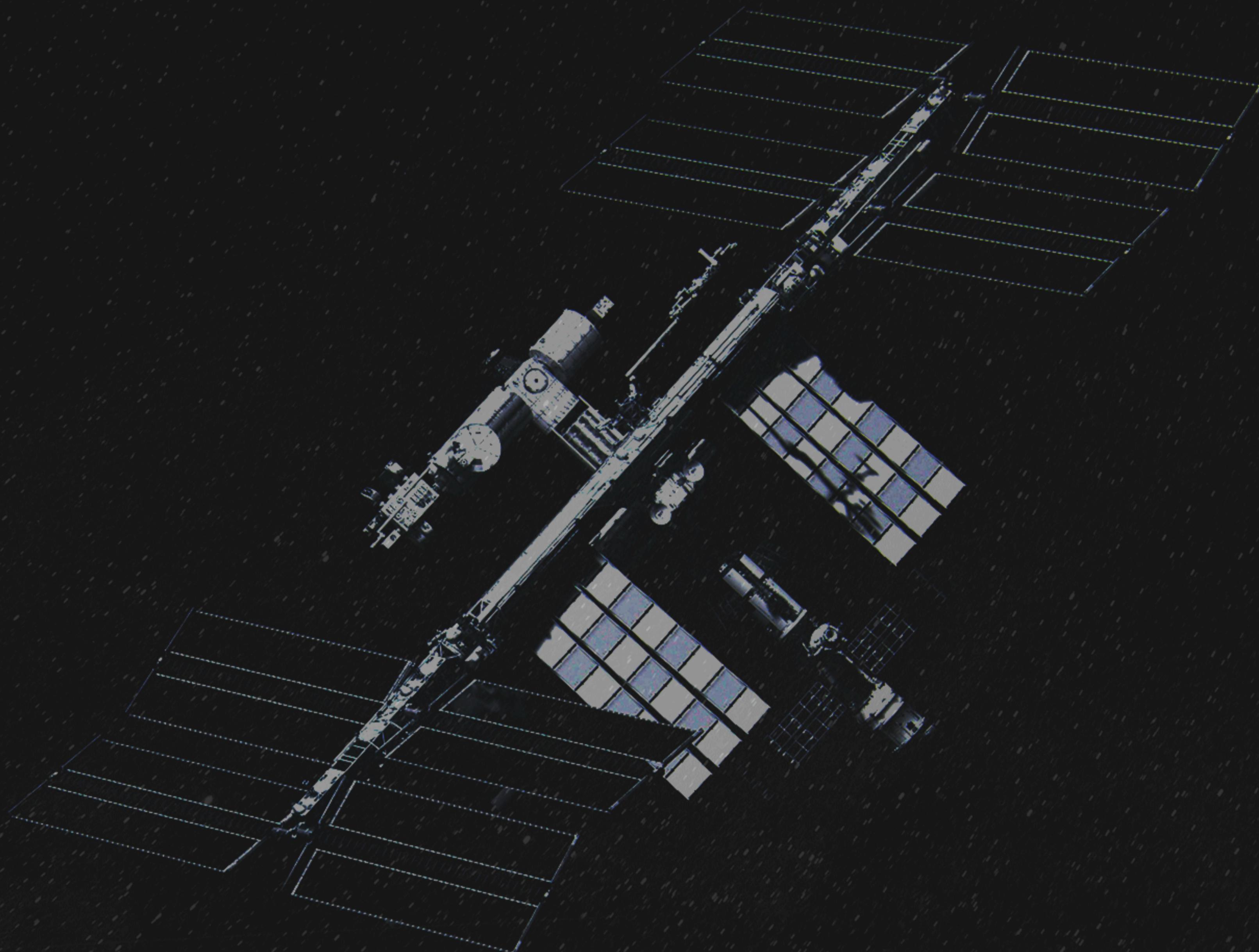
Saída: { '1': 0, '5': 0, '10': 0, '25': 2, '100': 0, '500': 0 }

Strings    Matemática

## 13. Código de Identificação de Nave

As naves da Federação utilizam um código único que serve para identificá-las. Esse código é composto de 12 algarismos, sendo o último um dígito verificador.

Você está construindo em sua própria nave um módulo capaz de analisar um código de identificação recebido e verificar se é um código válido de uma nave da Federação.



## Desafio 13: Código de Identificação de Nave

Escreva uma função que recebe um número e verifica se ele é um código de identificação válido segundo as regras de criação do dígito verificador. A criação do dígito verificador funciona da seguinte forma:

1. Some os dígitos das posições pares (ímpares se estiver contando a partir de 1)
2. Multiplique esse resultado por 3
3. Some os dígitos das posições ímpares (pares se estiver contando a partir de 1) do número original e então some esse resultado ao resultado do passo anterior
4. Encontre o resto da divisão do resultado do passo anterior por 10
5. Se o resto da divisão for 0, o dígito verificador é 0, do contrário o dígito verificador é  $10 - \text{resto}$

## Testes 13: Código de Identificação de Nave

- Entrada: (547020743789)

Saída: true

- Entrada: (301354030348)

Saída: true

- Entrada: (301354030349)

Saída: false

- Entrada: (123456789872)

Saída: false

Strings   Classes   Datas

## 14. Licença de Voo

Para pilotar uma nave de pequeno porte em qualquer planeta da Federação é preciso possuir uma licença de voo.

O código usado na licença de voo é criado a partir de informações de seu dono, como nome e data de nascimento.

Foi solicitado que você crie um método para gerar a licença de voo de uma determinada pessoa para ser integrado aos sistemas da Federação. É obrigatório fazer isso utilizando uma classe.

## Desafio 14: Licença de Voo

Escreva uma classe que contenha um método para gerar uma licença de voo e os seguintes atributos:

- Nome
- Sobrenome
- Data de Nascimento
- Licença de Voo (que deve iniciar sempre como falso)

Além disso a classe deve possuir um método para criar uma licença caso a pessoa ainda não possua uma. A licença deve ser uma string seguindo o seguinte padrão:

- Os primeiros cinco caracteres do sobrenome em letras maiúsculas (completado com 9's caso possua menos de cinco)
- O 6º caractere é um traço (-)
- O 7º caractere é o algarismo da década (penúltimo) do ano de nascimento
- O 8º e 9º caracteres são o mês de nascimento
- O 10º caractere é o algarismo do ano (último) do ano de nascimento
- O 11º caractere é um ponto (.)
- O 12º caractere é a primeira letra do primeiro nome (minúscula)

## Testes 14: Licença de Voo

```
- Pilot {  
    firstName: 'John',  
    lastName: 'Doe',  
    birthday: 1977-05-25T03:00:00.000Z,  
    flyingLicense: 'DOE99-7057.j'  
}  
- Pilot {  
    firstName: 'Hal',  
    lastName: 'Jordan',  
    birthday: 1995-09-02T03:00:00.000Z,  
    flyingLicense: 'JORDA-9095.h'  
}  
Pilot {  
    firstName: 'Carol',  
    lastName: 'Danvers',  
    birthday: 1968-08-17T03:00:00.000Z,  
    flyingLicense: 'DANVE-6088.c'  
}  
Pilot {  
    firstName: 'Poe',  
    lastName: 'Dameron',  
    birthday: 1979-03-09T03:00:00.000Z,  
    flyingLicense: 'DAMER-7039.p'  
}
```



**PARABÉNS!**

Você acaba de concluir  
**8 desafios Easy** e avançar  
na sua jornada pela galáxia.

**Sua nova atribuição:**

Programador(a) JavaScript  
das galáxias nível prata.

**onebitcode** {}

FOR AMAZING PROGRAMMERS

Tire um print, publique nos **stories do Instagram**  
e marque **@onebitcode**