

**ALUNO:** Jonas de Oliveira Castro Neto

**MATRÍCULA:** 2023111epdmd0086

**CURSO:** Programação Para Dispositivos Móveis

**Info.:** Os programas das questões abaixo devem ser codificados na linguagem Javascript

OBS.: Utilizarei um arquivo *index.html* que irá executar os scripts .js, conforme demonstrado abaixo, onde irei mudar somente o arquivo que irá ser executado conforme as questões:

```
Unidade3 > lista2 > index.html > ...
1  <!-- ESTE ARQUIVO IRÁ EXECUTAR OS SCRIPTS NO HTML -->
2  <!DOCTYPE html>
3  <html lang="pt-BR">
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Lista de Exercícios 2 da Unidade 3</title>
9  </head>
10 <body>
11     <!-- AQUI SERÁ CARREGADO OS SCRIPTS DAS QUESTÕES -->
12     <script src="q1.js"></script>
13 </body>
14 </html>
```

**Link do github**

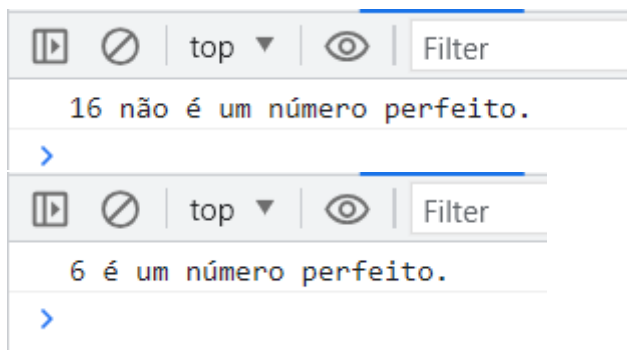
<https://github.com/2023111epdmd0086/logica-programacao/tree/main/Unidade3/lista2>

1. Faça um programa que leia um número inteiro e informe se ele é perfeito ou não. Um número perfeito é aquele que é igual à soma de seus divisores. Ex.:  $6 = 1 + 2 + 3 = \text{número perfeito}$

### Algoritmo

```
Unidade3 > lista2 > JS q1.js > somaDivisores
1  // Faça um programa que leia um número inteiro e informe se ele é perfeito ou não. Um
   número perfeito é aquele que é igual à soma de seus divisores. Ex.:  $6 = 1 + 2 + 3 =$ 
   número perfeito
2
3  const numero = parseInt(prompt("Digite um número inteiro:"));
4  let somaDivisores = 0;
5
6  // Percorre os possíveis divisores do número
7  for (let i = 1; i < numero; i++) {
8      if (numero % i === 0) {
9          somaDivisores += i; // Adiciona o divisor à soma
10     }
11 }
12
13 // Verifica se a soma dos divisores é igual ao número
14 if (somaDivisores === numero) {
15     console.log(`${numero} é um número perfeito.`);
16 } else {
17     console.log(`${numero} não é um número perfeito.`);
18 }
19
```

### Resultado



2. Faça um programa que solicite um número inteiro de até 4 dígitos ao usuário e inverta a ordem de seus algarismos. Ex.: Entrada = 5382 - Saída = 2835

### Algoritmo

```
Unidade3 > lista2 > JS q2.js > ...
1  // Faça um programa que solicite um número inteiro de até 4 dígitos ao usuário e
   // inverta a ordem de seus algarismos. Ex.: Entrada = 5382 - Saída = 2835
2
3  // Leitura do número digitado pelo usuário
4  const numero = prompt("Digite um número inteiro de até 4 dígitos:");
5
6  // Verifica se o número possui até 4 dígitos
7  if (numero.length <= 4) {
8      // Inverte a ordem dos algarismos utilizando o método split(), reverse() e join()
9      const numeroInvertido = numero.split("").reverse().join("");
10     console.log(`Número invertido: ${numeroInvertido}`);
11 } else {
12     console.log("O número deve ter até 4 dígitos.");
13 }
14
```

### Resultado

Essa página diz

Digite um número inteiro de até 4 dígitos:

Número invertido: 2835

>

3. Escreva um programa para verificar se um número é palíndromo (Número que é igual ao seu reverso Ex.: 14541)

#### Algoritmo

```
Unidade3 > lista2 > JS q3.js > ...
1 // Escreva um programa para verificar se um número é palíndromo (Número que é igual
  ao seu reverso Ex.: 14541)
2
3 // Leitura do número digitado pelo usuário
4 const numero = parseInt(prompt("Digite um número inteiro:"));
5
6 const numeroString = numero.toString(); // Converte o número em uma string
7
8 // separar em array, depois inverte, depois junta novamente
9 const numeroReverso = numeroString.split("").reverse().join("");
10
11 // Verifica se o número original é igual ao número reverso
12 if (numeroString === numeroReverso) {
13   console.log(`${numero} é um número palíndromo.`);
14 } else {
15   console.log(`${numero} não é um número palíndromo.`);
16 }
17
```

#### Resultado

Essa página diz

Digite um número inteiro:

top Filter

1456541 é um número palíndromo.

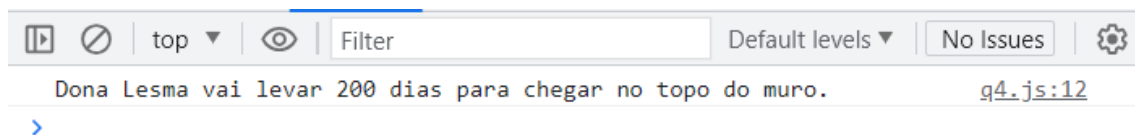
>

4. Dona Lesma é esportista e aventureira e definiu como objetivo deste verão alcançar o topo do muro do jardim em que vive. A cada dia, valente e metodicamente ela sobe exatamente uma certa distância (sempre a mesma a cada dia). Mas a cada noite enquanto dorme Dona Lesma escorrega para baixo uma outra distância (sempre a mesma a cada noite) ... Dadas a altura do muro, a distância que ela sobe a cada dia e a distância que ela desce a cada noite, ajude Dona Lesma a calcular quantos dias ela levará para chegar ao topo do muro. altura = 10000 subida = 100 descida = 50

### Algoritmo

```
1
2
3 // Definição das variáveis
4 const altura = 10000; // Altura total do muro
5 const subida = 100; // Distância que ela sobe a cada dia
6 const descida = 50; // Distância que ela desce a cada noite
7
8 // Cálculo do número de dias necessários para alcançar o topo do muro
9 const numeroDias = Math.ceil(altura / (subida - descida)); // Arredondando pra cima
10
11 // Exibição do resultado
12 console.log(`Dona Lesma vai levar ${numeroDias} dias para chegar no topo do muro.`);
13
```

### Resultado



5. Pedrinho está implementando o sistema de controle de pagamentos parcelados de uma grande empresa de cartão de crédito digital. Os clientes podem parcelar as compras sem juros no cartão, em até 18 vezes. Quando o valor  $V$  da compra é divisível pelo número  $P$  de parcelas que o cliente escolhe, todas as parcelas terão o mesmo valor. Por exemplo, se o cliente comprar um livro de  $V=30$  reais em  $P=6$  vezes, então as parcelas terão valores: 5, 5, 5, 5, 5 e 5. Mas se o valor da compra não for divisível pelo número de parcelas será preciso fazer um ajuste, pois a empresa quer que todas as parcelas tenham sempre um valor inteiro e somem no total, claro, o valor exato da compra. O que Pedrinho decidiu foi distribuir o resto da divisão de  $V$  por  $P$  igualmente entre as parcelas iniciais. Por exemplo, se a compra for de  $V=45$  e o número de parcelas for  $P=7$ , então as parcelas terão valores: 7, 7, 7, 6, 6, 6 e 6. Quer dizer, como o resto da divisão de 45 por 7 é 3, então as 3 parcelas iniciais devem ter valor um real maior do que as 4 parcelas finais. Você precisa ajudar Pedrinho e escrever um programa que, dado o valor da compra e o número de parcelas, imprima os valores de cada parcela. O programa deve receber como entrada o valor de  $V$ , representando o valor da compra e o valor de  $P$ , indicando o número de parcelas. A saída deve ser as parcelas.

### Algoritmo

```
Unidade3 > lista2 > JS q5js > ...
1 // Leitura do valor da compra e número de parcelas
2 const v = parseInt(prompt("Digite o valor da compra:")); // Valor da compra
3 const p = parseInt(prompt("Digite o número de parcelas desejado:")); // Número de parcelas
4
5 // Cálculo do valor das parcelas
6 const valorParcela = Math.floor(v / p); // Valor das parcelas sem o ajuste
7 const resto = v % p; // Resto da divisão
8
9 // Array para armazenar os valores das parcelas
10 const parcelas = [];
11
12 // Preenchimento do array com as parcelas
13 for (let i = 0; i < p; i++) {
14     parcelas.push(valorParcela); // Adiciona o valor das parcelas sem o ajuste
15     if (resto > 0) {
16         parcelas[i]++; // Incrementa 1 no valor das parcelas iniciais para ajustar o resto
17         resto--;
18     }
19 }
20
21 console.log("Valores das parcelas:");
22 console.log(parcelas);
23
```

### Resultado

Essa página diz

Digite o valor da compra:

OK

Cancelar

Essa página diz

Digite o número de parcelas desejado:

OK

Cancelar

Valores das parcelas:

▼ (10) [55, 55, 55, 55, 55, 55, 55, 55, 55, 55] ⓘ

0: 55

1: 55

2: 55

3: 55

4: 55

5: 55

6: 55

7: 55

8: 55

9: 55

length: 10

► [[Prototype]]: Array(0)