Lista 03

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

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

Imagine que temos o seguinte problema: Dado um número inteiro N, queremos encontrar todos os divisores positivos de N. Uma possível solução seria, novamente, iterar por todos os números de 1 a N e checar quais deles dividem N, salvando-os em um vetor de respostas. Dessa forma, obtemos uma complexidade de $O(N)$.

Para otimizar este algoritmo, perceba que se p é um divisor de N maior que $\sqrt{N}$, então o número $q = \frac{N}{p}$ também é um divisor de N e $q < \sqrt{N}$, Isso sugere que é suficiente iterarmos apenas pelos números de 1 a $\sqrt{N}$, e, assim que encontrarmos um divisor q tal que $q \leqslant \sqrt{N}$, inserimos tanto q quanto $\frac{N}{q}$ em nosso vetor de respostas. Dessa forma, encontraremos tanto os divisores ”pequenos” (menores ou iguais a $\sqrt{N}$) de N quanto os divisores ”grandes” (maiores que $\sqrt{N}$).

Essa versão otimizada do algoritmo reduz a complexidade de $O(N)$ para $O(\sqrt{N})$, tornando-o muito mais eficiente para números grandes.

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

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

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

dias	inicio do dia	fim do dia
1º	0	100
2º	50	150
3º	100	200
...	...	...
n	$(n-1) \times (\text{subida} - \text{descida})$	$(n-1) \times (\text{subida} - \text{descida}) + \text{subida}$

$$\text{distancia} = (n-1) \times (\text{subida} - \text{descida}) + \text{subida}$$

$$(n-1) \times (\text{subida} - \text{descida}) = \text{distancia} - \text{subida}$$

$$n - 1 = \frac{\text{distancia} - \text{subida}}{\text{subida} - \text{descida}}$$

$$n = \frac{\text{distancia} - \text{subida}}{\text{subida} - \text{descida}} + 1$$

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