

Desenvolvimento Front-end I

Ciclo 5 – Desafio JavaScript

Luccas Rafael

Organização dos arquivos do desafio

- Resolva cada exercício em um arquivo **separado**, em **JavaScript**.
- Sugestão de nomes de arquivos: `ex1.js`, `ex2.js`, ..., `ex10.js`.

Pontuação

Cada exercício vale **1,5** ponto, totalizando até **15** pontos.

Exercício 1 – Soma Simples

Descrição. Leia dois valores inteiros (variáveis A e B), calcule a soma e atribua à variável SOMA. Em seguida, escreva o valor desta variável.

Entrada. Dois valores inteiros.

Saída. Imprima a mensagem `SOMA = valor` (com um espaço antes e depois do sinal de igualdade) e finalize com quebra de linha.

```
let input = require("fs").readFileSync("/dev/stdin", "utf8");
let lines = input.split("\n");
let A = parseInt(lines.shift());
let B = parseInt(lines.shift());
```

Exercício 2 – Coordenadas de Um Ponto

Dica (leitura das variáveis A e B): Descrição. Leia dois valores com uma casa decimal (x e y) que representam as coordenadas de um ponto no plano e determine o quadrante a que pertence, se está sobre algum eixo ou se está na origem.

Entrada. As coordenadas de um ponto.

Saída. Uma mensagem indicando “Origem”, “Eixo X”, “Eixo Y” ou o quadrante correspondente.

```
var input = require('fs').readFileSync('/dev/stdin', 'utf8');
var lines = input.split('\n');
let [x, y] = lines.shift().trim().split(" ").map((x) => parseFloat(x));
```

Exercício 3 – Imposto de Renda

Dica (leitura em ponto flutuante): **Descrição.** Calcule e mostre o valor do Imposto de Renda devido conforme a tabela proposta.

Exemplo. Para salário de R\$ 3002,00, a taxa é de 8% sobre R\$ 1000,00 e 18% sobre R\$ 2,00, totalizando R\$ 80,36.

Entrada. Um valor de ponto flutuante com duas casas decimais.

Saída. Imprima R\$ seguido do valor devido com duas casas decimais. Se o valor de entrada for menor ou igual a 2000, imprima **Isento**.

Exercício 4 – Quadrante

Descrição. Leia repetidamente pares de coordenadas inteiras (X, Y). Para cada par, escreva o quadrante correspondente. O programa termina quando pelo menos uma das coordenadas for **nula** (neste caso não imprimir nenhuma mensagem).

Entrada. Vários casos de teste, cada um com dois inteiros.

Saída. O quadrante de cada coordenada lida.

Exercício 5 – Sequência Lógica 2

Descrição. Leia dois valores X e Y . Mostre a sequência de 1 até Y , quebrando a linha a cada X números.

Entrada. Dois inteiros com restrições $1 < X < 20$ e $X < Y < 100000$.

Saída. Cada linha contém X números separados por um espaço, sem espaço após o último número da linha.

Exercício 6 – Soma de Pares Consecutivos

Descrição. Leia um valor inteiro X indefinidas vezes (o programa para quando $X = 0$). Para cada X , imprima a soma dos cinco pares consecutivos a partir de X (incluindo X se ele for par).

Entrada. Muitos inteiros; o último é 0.

Saída. A soma dos cinco pares consecutivos para cada X .

Exemplos. Para $X = 4$ a saída é 40 ($4 + 6 + 8 + 10 + 12$); para $X = 11$ a saída é 80 ($12 + 14 + 16 + 18 + 20$).

Exercício 7 – Laércio

Descrição. Dada uma lista de inteiros, imprima apenas os números ímpares ordenados no padrão: maior, menor, segundo maior, segundo menor, e assim por diante.

Entrada. Um inteiro N ($1 < N < 1000$) indicando o número de casos. Para cada caso, um inteiro M ($0 < M < 100$) seguido de M inteiros M_i ($0 < M_i < 1000$).
Saída. A lista ordenada no formato solicitado; um caso por linha.

Exercício 8 – Uma Jornada Inesperada

Descrição. Calcule em quantos dias o grupo chega à Montanha Solitária. O tempo é a distância (em km) dividida pela quantidade de pessoas na jornada, que é o número de anões somado a 2 (Bilbo e Gandalf).

Entrada. Dois inteiros N (número de anões) e X (distância).

Saída. Um número real com duas casas decimais representando os dias necessários.

Exercício 9 – Grupo de Trabalho do Noel

Descrição. Cada elfo pertence a um dos quatro grupos de trabalho, com as seguintes capacidades por presente: bonecos (8h), arquitetos (4h), músicos (6h), desenhistas (12h). Dada a lista de elfos com grupo e horas que irão trabalhar por dia, calcule quantos presentes por dia serão produzidos.

Entrada. Um inteiro N ($1 \leq N \leq 1000$) seguido de N linhas com E (nome), G (grupo em minúsculas) e H (horas por dia, $1 \leq H \leq 24$).

Saída. Um inteiro P indicando o total de presentes por dia.

Exercício 10 – Faixa de Letras

Definição. Uma *faixa de letras* é um conjunto de letras minúsculas consecutivas (“a” a “z”). Representa-se por `menor:maior`. Exemplos: `a:c` representa $\{a,b,c\}$; `w:z` representa $\{w,x,y,z\}$; `m:m` representa $\{m\}$.

Entrada. Cada linha é um caso de teste: uma string (0–50 caracteres) formada por letras minúsculas e espaços, possivelmente fora de ordem.

Saída. Para cada caso, imprimir as faixas de letras ordenadas pelo menor elemento e cobrindo a maior sequência possível em ordem crescente. Ignorar espaços e duplicatas. Se a string for vazia, imprimir apenas uma linha em branco.

Exemplo. Para a string `"fb xee ac"`, as faixas são: `a:c`, `e:f`, `x:x`.

O que deve ser entregue?

- Um arquivo compactado no formato **ZIP**.
- Dentro do arquivo ZIP devem estar todos os exercícios resolvidos.
- Cada exercício deve estar em um arquivo **separado** e nomeado de forma clara (por exemplo: `ex1.js`, `ex2.js`, `ex3.js`, ...).
- As soluções devem ser implementadas em **JavaScript**.

Como vai ser a avaliação?

- A correção será feita executando os arquivos enviados, verificando entrada/saída conforme os enunciados.
- Critérios: **correção da lógica, aderência ao formato de saída, organização do código e nomeação dos arquivos.**
- Entregas fora do padrão de nomeação/estrutura poderão perder pontos.