



A maior Olimpíada de Algoritmo entre  
alunos de ensino técnico do Brasil

**Prezado(a) competidor(a),**

Parabéns! Chegar até a Final de uma Olimpíada tão concorrida já é motivo de orgulho para sua escola. Agora você faz parte de um grupo seleta entre os melhores aspirantes a programadores do Brasil. A Olimpíada de Algoritmo da Hostnet pretende ser um impulso para que você, estudante, voe mais alto. Nosso objetivo é incentivar o estudo da matéria e promover o desenvolvimento do software no país.

#### **Sobre a prova:**

Esta prova foi formulada com base no curriculum de Algoritmos definido pelo MEC e ensinado nas escolas técnicas de nível médio. Os desafios exploram a habilidade do competidor dentro do conteúdo aprendido.

A interpretação das questões bem como sua compreensão fazem parte da avaliação, por isso não será permitido o esclarecimento de dúvidas por professores ou fiscais de prova.

Notas básicas sobre a avaliação que devem ser destacadas:

- Os algoritmos são avaliados não apenas pelo seu resultado (saída), mas também pela sua baixa complexidade e alta qualidade de execução.
- A organização e a legibilidade do código também são levadas em consideração.
- Os algoritmos deverão ser feitos em código genérico estruturado, ou seja, em pseudocódigo (também conhecido como português), fazendo uso das respectivas estruturas fundamentais do algoritmo estruturado, listadas no conteúdo programático, não sendo permitido algoritmos com representação gráfica, narrativa textual, ou qualquer outra representação não estruturada.

Fique tranquilo(a), divida seu tempo e leia os enunciados com atenção. Sua resposta deve conter o algoritmo mais rápido e otimizado possível para responder à questão. A organização e a legibilidade do código também serão levados em consideração. Só serão aceitos algoritmos em pseudocódigo e não algoritmos em representação gráfica, como fluxogramas estruturados.

**Boa sorte!**

Kauê Linden

Diretor da Hostnet

**REALIZAÇÃO**



**www.oah.com.br**



# Prova Final Nacional

A criptografia é um processo usado para embaralhar dados que só podem ser recuperados por um processo inverso para desembaralhar. É muito utilizado em dados sigilosos, especialmente na informática.

Embora pareça uma técnica nova, a criptografia é muito antiga. Sabe-se que imperadores romanos já utilizavam técnicas de criptografia para comunicação militar. A técnica normalmente usada por eles era a cifra de troca ou cifra de César.

A cifra de César consiste em trocar as letras de um texto por outras localizadas passos a frente. Por exemplo, uma cifra de César com passo 1, trocava cada letra A por uma B, cada letra B por uma C e assim por diante. A palavra CASAMENTO viraria DB-TBNFOUP na cifra com passo 1.

Esse tipo de criptografia, atualmente, não é segura para proteger informações sigilosas, porém algumas ainda são utilizadas para outros fins. Vejamos o exemplo do ROT13.

O ROT13 é uma cifra de César de 13 passos, utilizada para o alfabeto padrão inglês. É a mais utilizada quando se quer proteger textos de mensagens na internet. Por exemplo, o usuário pode conversar em fóruns de discussão ou enviar mensagens usando ROT13 para que outras pessoas não consigam ler o texto facilmente. É muito útil quando se quer esconder comentários reveladores sobre filmes ou livros em fóruns, entre outras coisas.

Existem muitos decodificadores de ROT13 disponíveis na internet.

## O problema

Está em desenvolvimento uma ferramenta que necessitará de funções diversas para uso com textos. Dentre todas as funções a desenvolver, faça os algoritmos das seguintes:

a) Uma função que receba um texto e o retorne codificado em ROT-13 (alfabeto da língua inglesa).

b) Uma função que receba um número inteiro positivo com qualquer número de caracteres e retorne o número formatado com os pontos de milhar. Ex: 1250 vira 1.250, 58963 vira 58.963, 78852557 vira 78.852.557.

c) Uma função que receba dois textos. Um texto comum e um texto que seria uma palavra ou uma cadeia de caracteres a ser buscada no outro texto. A função deve retornar quantas vezes a palavra procurada aparece no texto. Ex: Procurando a cadeia "te" no texto "Nem teme, quem te adora, a própria morte", a função retornará 3.

d) Uma função que receba um texto e o retorne com suas palavras em ordem alfabética pela primeira letra. Ex: "Pato cão gato" vira "cão gato Pato".

Algumas funções já estão prontas, podendo ser úteis para desenvolver as funções acima. Use as em qualquer

parte dos algoritmos se precisar. Elas são as seguintes:

**TamanhoTexto(Texto : texto) : inteiro**

Recebe um texto e retorna o seu tamanho.

Ex: TamanhoTexto("Olimpíada de Algoritmo") retorna 22.

**VetorCaracteres(Texto : texto) : vetor**

Recebe um texto e retorna um vetor, onde cada elemento do vetor será uma letra do texto.

Ex: VetorCaracteres("Formação técnica.") retorna o vetor abaixo com o índice começando do número zero.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
F	o	r	m	a	ç	ã	o		t	é	c	n	i	c	a	.

**Aloca(Vetor : vetor, Qtd : inteiro)**

Serve para alocar mais espaços em um vetor. Com ele, pode-se iniciar um vetor no algoritmo com 1 elemento e depois, usando a função, é possível alocar mais espaços conforme necessitar.

Ex: Aloca(VTexto, 5) adiciona mais 5 espaços vazios no final do vetor VTexto.

REALIZAÇÃO

APÓIO

