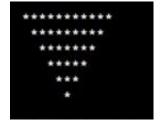


LISTEX 002

Participe do grupo do WhatsApp para ajudar na resolução. Além disso, o grupo é benéfico para fazer networking! O Link do grupo está no módulo de boas-vindas!

LEIA O DOCUMENTO ATÉ O FIM. TENHO ALGUNS AVISOS IMPORTANTES A DAR.

- 1) Um professor está ensinando lógica proposicional aos seus alunos e para isso pediu o seu auxílio para criar um algoritmo que fosse capaz de converter lógica proposicional em valores finais (verdadeiro/falso). Desta forma, quando o usuário digitar uma expressão (ex.: ~P v Q), o sistema deverá ler a expressão, realizar os cálculos e apresentar o resultado lógico (V ou F).
 - a) Dica: Leia um caractere de cada vez ao invés de pedir a expressão inteira.
 - b) Dica2: Pergunte ao usuário a quantidade de caracteres a ser lido antes de iniciar
 - c) Dica3: Foque em expressões mais simples, com no máximo 2 proposições, para facilitar.
- 2) Crie um algoritmo que gere o seguinte desenho na tela:





- 3) Crie um algoritmo com laços de repetição e vetores que descubra o N-ésimo valor da sequência de Fibonacci conforme solicitado pelo usuário. Ex.: O usuário deverá pedir o 168º número da sequência e o sistema deverá informar qual número é este.
- 4) Crie uma função recursiva que seja capaz de calcular o N-ésimo valor da sequência de Fibonacci conforme solicitado pelo usuário. Ex.: O usuário deverá pedir o 168° número da sequência e o sistema deverá informar qual número é este.

A programação é um mar, ou melhor, um oceano. Existe MUITA COISA ainda que você pode explorar dentro de Lógica de Programação. Recomendo que dê uma pesquisada nos itens a seguir; garanto que eles farão de você um programador muito mais eficiente que a maioria!

- a) Algoritmos de busca linear (modo lento dependendo da entrada)
- b) Algoritmos de busca binária (modo rápido dependendo da entrada)
- c) Algoritmos de ordenação mais utilizados (tente implementar no VisuAlg!):
 - i) Quicksort
 - ii) Bubblesort
 - iii) Mergesort
 - iv) Selectionsort
- d) Estrutura de dados
 - i) Lista
 - 1) Fila
 - 2) Pilha
 - 3) Lista Encadeada
 - 4) Lista Duplamente encadeada



- ii) [aqui entraria vetores também, mas já vimos em aula]
- e) Complexidade de algoritmos (Se você gosta de matemática, irá se apaixonar por este tema, que toma a matemática como base para descobrir e comparar velocidades de execução de algoritmos).

Esta é a segunda e última lista de exercícios.

Você pode, todavia, chamar a qualquer momento no WhatsApp para tirar dúvidas sobre programação, inclusive, quando iniciar em alguma linguagem, recomendo fortemente que me chame para que receba eventuais orientações, combinado?

AVISO IMPORTANTE

Alguns cursos recomendados para você dar continuidade nos estudos:

- Python do junior ao Senior: https://www.logicamentepro.com.br/python/
- PHP Developer: https://bit.ly/curso-completo-php-dev
- JavaScript FullStack: https://bit.ly/bootcampo-javascript
- Curso de Inglês c/ professora nativa: https://bit.ly/aprendaInglesComGringa
- Curso de C# (Criação de softwares): https://bit.ly/curso-completo-c-sharp

Um abraço,

Prof. Márcio Santos

www.marciosantos.com.br

www.codigoceo.com.br

Instagram: @prof.msantos

@CodigoCEO