Universidade Federal do Piauí – UFPI

Campus Senador Helvídio Nunes de Barros – CSHNB Curso de Sistemas de Informação Bloco: IV

Disciplina: Estruturas de Dados II Professora: Juliana Oliveira de Carvalho

Acadêmico:	Matrícula:
Academico	

TRABALHO DE ESTRUTURAS DE DADOS II PARA A SEGUNDA AVALIAÇÃO

- 1) **(2,0 pontos)** Faça um programa em C de referência cruzada que construa uma árvore binária de busca vermelha-preta, com todas as palavras incluídas a partir de um arquivo texto, e armazene o número de cada linha em que a palavra foi usada. Os números das linhas devem ser armazenados em uma lista encadeada associada ao nó da árvore. Depois do arquivo de entrada ter sido processado, imprima em ordem alfabética todas as palavras do arquivo texto, junto com os números das linhas onde foi usada. O programa deve permitir o usuário buscar uma palavra e o programa deve responder em quais linhas do texto ela foi utilizada, para cada palavra buscada mostre o número de passos até alcançar a palavra na árvore. Além disso, ele deve permitir excluir uma palavra de uma linha bem como acrescentar.
- 2) **(3,0 Pontos)** Repita o exercício 1 para uma árvore 2-3. obs.: No relatório compare o tempo de inserção e de busca na árvore do exercício 1 e do 2.
- 3) **(5,0 pontos)** Suponha que uma memória seja dividida em blocos lógicos de 1Mbyte, e que o primeiro bloco é o bloco 0, suponha também que o gerenciador de memória de um Sistema Operacional mantenha uma árvore 2-3 com nós para blocos livres e ocupados da memória. Cada nó da árvore 2-3 contém os seguintes campos: O- para ocupado ou L- para livre, o número do bloco inicial, o número do bloco final, endereço inicial (correspondente ao endereço inicial do primeiro bloco do nó) e endereço final (correspondente ao endereço final do último bloco do nó). Os nós da árvore será organizados de forma que um nó ocupado está entre 2 nós livres e vice-versa.
 - (a) Faça um programa em C que cadastra os nós da árvore, onde o usuário deve informar se o primeiro nó é livre ou ocupado, o endereço inicial e final do nó. Os demais nós serão contabilizados pelo sistema se são livres ou ocupados e o usuário deve apenas informar o endereço inicial e final de cada um. O cadastro termina quando o usuário informar como endereço final de um nó o último endereço da memória.
 - (b) Faça uma função que o usuário informe a quantidade de blocos que ele precisa alocar e retorne as informações do nó que atenda as necessidades do usuário e então modifique a situação do referido nó de Livre para Ocupado.
 - (i) Lembre-se que a árvore deve manter nós intercalados de acordo com a situação do nó, ou seja, se a situação de um nó muda então os nós adjacentes a ele deve ser concatenados. Consequentemente os nós da árvore serão modificados.
 - (ii) Se o usuário informar uma quantidade de blocos que não existe livre, diga ao usuário que não é possível alocar aquela quantidade de blocos.
 - (c) Faça uma função em que o usuário informe blocos (dizendo o número do primeiro e do último bloco) que devem ser liberados.
 - (i) Lembre-se que a árvore deve manter nós intercalados de acordo com a situação do nó, ou seja, se a situação de um nó muda então os nós adjacentes a ele deve ser concatenados. Consequentemente os nós da árvore serão modificados.
 - (ii) Se o usuário informar um número de blocos que não estejam ocupados, informe ao usuário que a sequência informada não está ocupada.

Equipe: os programas podem ser feitos em dupla, mas os relatórios são individuais. Se os programas forem feitos em dupla, a dupla deve ser identificada no envio do código.

Data de Entrega: agendada no Sigaa

Entregar: Código Fonte, Relatório(Conforme Modelo em PDF)

Forma de Entrega: pelo SIGAA, caso tenha algum problema enviar por e-mail(julianaoc@gmail.com).

Entrevista Individual: agendar horário com a Professora.

Obs.: Todos os alunos devem entregar o código e relatório, mesmo que o código esteja igual a do outro membro da dupla.