

### TRABALHO DE ESTRUTURAS DE DADOS II PARA A PRIMEIRA AVALIAÇÃO

- 1) **(3,0 pontos)** Faça um programa em C que cadastre em uma árvore binária os dados de:
  - (a) Curso: Código, nome do curso, quantidade de blocos, número de semanas para cada disciplina (exemplo: 15, 18, 20. Na UFPI é 15) e o endereço para a árvore de disciplinas daquele curso. A árvore deve estar organizada pelo código do curso, o mesmo não pode se repetir e cada curso terá uma árvore de disciplinas;
  - (b) Disciplinas: Código da disciplina, nome da disciplina, bloco da disciplina(o qual deve ser menor que a quantidade de blocos do curso), a carga horária da disciplina(deve ser um número múltiplo do número de semanas para cada disciplina no curso). A árvore deve estar organizada pelo código da disciplina, o mesmo não deve se repetir;
  - (c) Depois permita ao usuário:
    - (1) Imprimir a árvore de cursos em ordem crescente pelo código do curso;
    - (2) Imprimir os dados de um curso dado o código do mesmo;
    - (3) Imprimir todos os cursos com a mesma quantidade de blocos, onde o usuário informe a quantidade de blocos;
    - (4) Imprimir a árvore de disciplinas em ordem crescente pelo código das disciplinas dado o código do curso;
    - (5) Imprimir os dados de uma disciplina dado o código dela e do curso ao qual ela pertence;
    - (6) Imprimir as disciplinas de um determinado bloco de um curso, dado o bloco e o código do curso;
    - (7) Imprimir todas as disciplinas de um determinado curso com a mesma carga horária, onde o código do curso e a carga horária devem ser informadas pelo usuário;
    - (8) Excluir uma disciplina dado o código da disciplina e o código do curso;
    - (9) Excluir um curso dado o código do mesmo, desde que não tenha nenhuma disciplina cadastrada para aquele curso.
- 2) **(2,0 pontos)** Em relação ao exercício 1, verifique os tempos:
  - (a) de inserção de cada elemento na árvore de cursos (no item a do exercício 1);
  - (b) de busca de um determinado curso na árvore de curso (no item (c.2) do exercício 1);

Para verificar os tempos de execução pesquise pela função time do C. Insira no código comandos para obter o tempo inicial antes de fazer a pesquisa e o tempo final depois de concluir a pesquisa.

Obs. 1: para os experimentos cada execução deve inserir os mesmos códigos de cursos em ordem diferente (pode utilizar comando para embaralhar os códigos e assim as árvores ficarem bem aleatórias e poder verificar a diferença entre inserções).

Obs. 2: Lembre-se que não pode haver impressão entre o tempo inicial e o tempo final, pois impressão consome muito tempo.

Obs. 3: Para validar o tempo de busca, o mesmo deve ser repetido 30 vezes. Faça uma média para obter o resultado final.

- 3) **(5,0 pontos)** Repita todo o processo dos exercícios 1 e 2 mas agora implemente o código usando uma Árvore AVL.

**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@ufpi.edu.br).

**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.**