



**UFAM**

**Universidade Federal do Amazonas  
Instituto de Computação**

**Algoritmos e Estruturas de Dados 2 (AED2)  
Trabalho Prático 1 - entrega até 08 e 10/04/2018**

Este primeiro trabalho prático de AED 2 tem como objetivo compreender a complexidade de diferentes algoritmos de busca e ordenação usando as estruturas de vetores e listas encadeadas. A implementação deve ser feita usando a linguagem C (obrigatório). Também deve ser implementado segundo os conceitos de Tipos Abstratos de Dados (obrigatório).

1. Desenvolva um programa que implemente um vetor com 1.000.000 (um milhão) de elementos do tipo inteiro, ordenado com valores aleatórios. Execute a busca sequencial e a busca binária 30 vezes usando os mesmos valores de busca para os dois algoritmos em cada execução. Calcule e mostre o tempo de execução para cada busca e a média entre as 30 execuções para cada algoritmo.
2. Aproveitando o vetor da questão anterior, crie uma lista encadeada com os mesmos valores. Execute a busca sequencial 30 vezes no vetor e 30 vezes na lista encadeada, usando os mesmos valores para busca em cada execução. Registre os tempos de execução de cada busca, calcule e exiba a média dos tempos de execução para cada estrutura.
3. Desenvolva um programa que implementa um vetor com 100.000 (cem mil) elementos do tipo inteiro, com valores aleatórios desordenados. Ordene o vetor usando três técnicas: Bolha, Inserção Direta e *Quicksort*. Execute cada algoritmo de ordenação 30 vezes, gerando novos valores para o vetor a cada uma das 30 rodadas. Em cada rodada, use o mesmo vetor (com os mesmos valores) para cada uma das três técnicas de ordenação. Calcule e mostre o tempo para cada execução de cada algoritmo de ordenação e a média entre as 30 execuções para cada algoritmo.
4. Observe o uso de memória do seu programa por meio de um gerenciador de processos (tarefas). Anote as variações significativas de ocupação da memória ao longo da execução do programa. Qual o menor e maior tamanho do programa? Em que momento ocorre as maiores variações de ocupação da memória. Coloque pausas (leitura do teclado, por exemplo) para facilitar essa tarefa.