

Ferramenta Computacional para o Ensino de Algoritmos de Ordenação

Rodrigo M. S. Veras¹, Flavio H. D. Araujo¹, Romuere R. V. Silva¹, Iális C. de Paula Jr²

¹Departamento de Informática e Estatística, Universidade Federal do Piauí (UFPI)
Teresina – PI – Brasil

²Engenharia da Computação, Universidade Federal do Ceará - Campus de Sobral (UFC)
Sobral – CE – Brasil

{rveras, flavio86, romuere}@ufpi.edu.br, ialis@ufc.br

Abstract. *Sorting is one of the most fundamental algorithmic problems within computer science. It has been claimed that as many as 25% of all CPU cycles are spent sorting, which provides a lot of incentive for further study. In addition, there are many other tasks (searching, calculating the median, remove duplicates, etc.) that can be implemented much more efficiently once the data is sorted. The process of algorithms and programming learning is considered a challenge for Computer Science students. This paper presents a tool to support the teaching of sorting algorithms that is facilitating the learning of the topic.*

Resumo. *Ordenação é um dos problemas algorítmicos fundamentais em Computação. Cerca de 25% de todos os ciclos da CPU são gastos com ordenação de dados, o que incentiva estudos mais aprofundados do tema. Além disso, há um grande número de outras tarefas (pesquisa, cálculo da média, remoção de dados duplicados, etc.) que podem ser implementadas mais eficientemente caso os dados estejam ordenados. Contudo, o processo de aprendizagem destes algoritmos é considerado um desafio para estudantes de Computação. Este trabalho apresenta uma ferramenta de apoio ao ensino de algoritmos de ordenação que vem facilitando o aprendizado do tema em questão.*

1. Introdução

O estudo de algoritmos e estruturas de dados é uma área fundamental no campo da Ciência da Computação, Engenharia da Computação, Bacharelado em Sistemas de Informação, entre outros cursos. Cada vez mais a necessidade de alta performance de sistemas computacionais torna-se indispensável tanto para um bom funcionamento, quanto para sua aceitação no mercado. Todavia, a aprendizagem de algoritmos e estruturas de dados constitui um dos grandes problemas enfrentados pelos alunos que ingressam nestes cursos já que as disciplinas que abordam tais conhecimentos estão entre as que possuem maior índice de reprovação [Barcelos and Tarouco 2009].

Com o intuito de produzir melhores resultados no processo de aprendizagem em cursos da área de Computação, faz-se constante a necessidade de atualização das didáticas de ensino de forma geral, procurando transformar processos abstratos em concretos através do uso de *software* e possibilitando melhor compreensão dos tópicos abordados e maior interação entre o aluno e objeto de trabalho [Santos et al. 2008]. As Tecnologias da Informação e Comunicação auxiliam a educação na relação do ensino com a

aprendizagem [Amen and Nunes 2006] e vem forçando que educadores e pesquisadores utilizem novos métodos e paradigmas na educação.

Nesse trabalho apresentamos a primeira versão do SEED (*Software de Ensino de Estruturas de Dados*). No estágio atual, o SEED provê ao aluno meios de estudo e comparação dos principais algoritmos de ordenação de dados e pode ser utilizado como ferramenta de ensino-aprendizagem nas disciplinas Estruturas de Dados e Projeto e Análise de Algoritmos (PAA).

2. Trabalhos Relacionados

Há diversas ferramentas construídas para o ensino de algoritmos e programação. Cada trabalho apresenta um foco relacionado às necessidades de seus desenvolvedores e professores interessados. Podem ser citados: *ASTRAL (Animation of Data Structures and Algorithms)* [Garcia et al. 1997], *EDDL (Estruturas de Dados Dinâmicas Lineares)* [Azul and Mendes 1998], *TBC-AED (Treinamento Baseado em Computador para Algoritmos e Estruturas de Dados)* [Santos and Costa 2005] e *TBC-GRAFOS (Treinamento Baseado em Computador para Algoritmos em Grafos)* [Santos and Costa 2006].

De maneira geral, essas ferramentas consistem em ambientes nos quais é possível a visualização gráfica da atuação de algoritmos sobre as estruturas de dados. Seus objetivos são apresentar uma abordagem inicial dos algoritmos e mostrar suas técnicas de implementação.

Na disciplina de PAA, os alunos da área de Computação devem aprender as principais técnicas de programação (divisão e conquista, algoritmos gulosos e programação dinâmica) e a analisar a complexidade assintótica dos algoritmos. A complexidade de um algoritmo pode ser definida com uma medida do seu desempenho que, na maioria dos casos, é dada em função do tempo gasto para se chegar à solução. Dessa forma, faz-se necessário a inclusão de um módulo para que o aluno possa visualizar e comparar, na prática, o desempenho dos algoritmos estudados.

3. Apresentação da Ferramenta

O SEED, foi desenvolvido utilizando-se a linguagem de programação Java (visando a portabilidade e futura utilização na *web*) e o ambiente de desenvolvimento *NetBeans*. Em sua primeira versão, a ferramenta possibilita o ensino-aprendizagem dos principais métodos de ordenação:

1. Ordenação por Inserção (*Insertion Sort*);
2. Ordenação por Seleção (*Selection Sort*);
3. Ordenação pelo método da bolha (*BubbleSort*);
4. Ordenação por Intercalação (*MergeSort*);
5. Ordenação por Montes (*HeapSort*);
6. Ordenação Rápida (*QuickSort*).

O SEED é dividido em dois módulos: um de aprendizagem e outro de comparação. No módulo de aprendizagem, o usuário tem a possibilidade de escolher um dos métodos disponíveis e poderá visualizar as suas principais informações: princípio do método, pseudocódigo e análise de complexidade. Além disso, o usuário pode optar

por executar o algoritmo, escolhendo a quantidade e a disposição (ordem crescente, decrescente ou aleatória) dos elementos, e visualizar o tempo de execução e a quantidade de comparações realizadas. A Figura 1 apresenta o módulo de aprendizado do SEED.

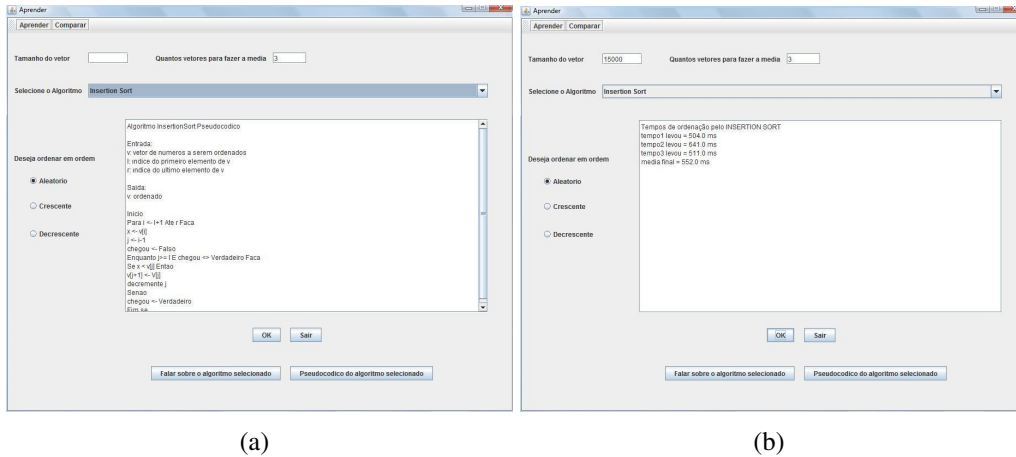


Figura 1. Visão do módulo de aprendizado do SEED: (a) visualização das informações do algoritmo, (b) resultado da execução de algoritmo.

No módulo de comparação, o usuário tem a possibilidade de avaliar o desempenho de dois ou mais métodos de ordenação. Ele pode selecionar quais métodos deseja comparar para uma determinada disposição dos dados. Selecionada essas duas informações, o SEED analisará o desempenho dos algoritmos selecionados para entradas de diferentes tamanhos: 1000, 5000, 10000, 20000 e 30000 elementos.

Os resultados são apresentados em formato de tabela e gráfico relacionando o tempo ou a quantidade de comparações com a quantidade de elementos a ser ordenado. Dessa forma, o usuário poderá visualizar qual método funciona melhor para uma determinada entrada de dados. Além disso, a representação gráfica dos resultados fornece uma visualização do comportamento assintótico (complexidade) dos algoritmos. A Figura 2 apresenta o módulo de comparação do SEED.

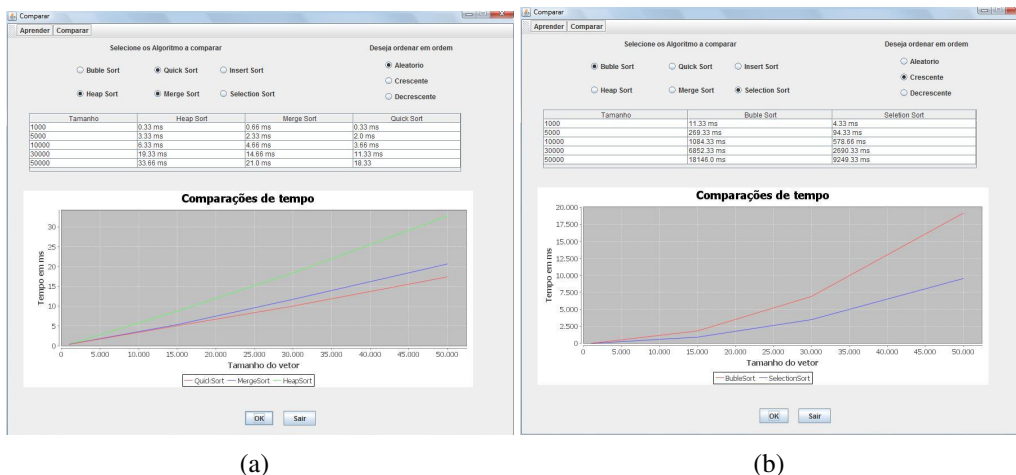


Figura 2. Visão do módulo de comparação do SEED: escolha e resultado da comparação de algoritmos.

4. Conclusões e Trabalhos Futuros

Para minimizar as dificuldades de ensino-aprendizagem, novas metodologias vêm sendo estudadas, principalmente no que diz respeito ao uso de recursos da tecnologia na educação. Dessa forma, estamos apresentando a primeira versão do SEED, um *software* que tem como principal objetivo o auxílio ao aluno de computação na compreensão dos métodos de ordenação.

Os testes preliminares foram realizados na disciplina de PAA oferecida pelo Departamento de Informática e Estatística ao curso de Bacharelado em Ciência da Computação na Universidade Federal do Piauí, na cidade de Teresina. A ferramenta foi utilizada nas atividades de monitoria e no auxílio ao ensino em PAA. Seu uso se mostrou satisfatório e foi bem aceito pelos alunos.

Durante o período de uso da ferramenta, foi observado um aumento do interesse e da compreensão dos alunos no estudo dos algoritmos de ordenação. Portanto, a ferramenta pedagógica desenvolvida apresenta um impacto positivo no aprendizado do referido tema.

Esta versão inicial da ferramenta foi desenvolvida para ser uma aplicação *standalone*. No entanto, futuramente possuirá recursos *web*, que possibilitem o acesso remoto, além de permitir troca de idéias, através da interatividade com usuários diversos. Versões futuras incluirão mais algoritmos clássicos em Ciência da Computação como algoritmos de busca em listas e algoritmos de manipulação de árvores e grafos.

Referências

- Amen, B. and Nunes, L. (2006). Tecnologias de informação e comunicação: contribuições para o processo interdisciplinar no ensino superior. *Revista Brasileira de Educação Médica*, 30(3):171–180.
- Azul, A. A. and Mendes, A. J. (1998). Eddl: Um programa didático sobre estruturas de dados dinâmicas lineares. In *3º Simpósio Investigação e Desenvolvimento de Software Educativo*, Évora, Portugal.
- Barcelos, R. J. and Tarouco, L. (2009). O uso de mobile learning no ensino de algoritmos. *RENOTE*, 7(3).
- Garcia, I. C., Rezende, P. J., and Calheiros, F. C. (1997). Astral: Um ambiente para ensino de estruturas de dados através de animações de algoritmos. *RBIE*, (1).
- Santos, R. P. and Costa, H. A. X. (2005). Tbc-aed e tbc-aed/web: Um desafio no ensino de algoritmos, estruturas de dados e programação. In *Anais do IV WEIMIG*, Varginha/MG.
- Santos, R. P. and Costa, H. A. X. (2006). Um software gráfico educacional para o ensino de algoritmos em grafos. In *Proc. of the IADIS/CIAWI 2006*, pages 358–362, Murcia/Espanha.
- Santos, R. P., Costa, H. A. X., Resende, A. M. P., and Souza, J. M. (2008). O uso de ambientes gráficos para ensino e aprendizagem de estruturas de dados e de algoritmos em grafos. In *Anais do XXVIII Congresso da SBC - WEI*, pages 157–166, Belém/PA.