

AESDA: Ferramenta Educacional Gráfica Extensível para Ensino de Algoritmos de Ordenação e Pesquisa com Ênfase na Análise da Eficiência de Algoritmos

Murillo Silva, Paulo Afonso Parreira Júnior

Universidade Federal de Goiás (UFG)
Jataí – GO – Brasil

`murillofnc@gmail.com, paulojunior@jatai.ufg.br`

***Abstract.** The literature reports difficulties encountered by students in understanding the abstract concepts of certain subjects, such as Algorithm and Data Structure (ADS). A proposal to minimize this problem is the use of graphical educational tools. Many tools have been developed with a focus on teaching ADS, but with certain limitations. So, the aim of this paper is to present the AESDA, an extensible educational tool for teaching sort and search methods with emphasis on analysis of the efficiency of algorithms. The main difference between AESDA and others tools existing in the literature is the fact that its functionality can be extended, allowing the new type of algorithms can be added to it. This article also presents an evaluation of AESDA that got as result 95% of acceptance regards to ease of use and 85% to the utility of this tool.*

***Resumo.** A literatura relata dificuldades encontradas por alunos em compreender os conceitos abstratos de certas disciplinas, como Algoritmo e Estrutura de Dados (AED). Uma proposta para tentar minimizar esse problema é a utilização de ferramentas gráficas de apoio. Muitas ferramentas têm sido desenvolvidas com enfoque no ensino de AED, porém com certas limitações. Sendo assim, o objetivo deste trabalho é apresentar a AESDA, uma ferramenta educacional extensível para ensino de métodos de ordenação e pesquisa com ênfase na análise da eficiência de algoritmos. O maior diferencial da AESDA está no fato de que ela pode ser estendida, de forma que novos tipos de algoritmos possam ser adicionados a sua funcionalidade. Este artigo apresenta ainda uma avaliação qualitativa da AESDA, que obteve como resultados 95% de aceitação com relação à facilidade de uso e 85% quanto à utilidade dessa ferramenta.*

1. Introdução

O mercado de trabalho em Tecnologia da Informação e Comunicação (TIC) está cada vez mais exigente, requisitando que os profissionais sejam formados com bom conhecimento nas mais diversas áreas da computação. Desta forma, é importante que o processo de ensino-aprendizagem seja o mais adequado e homogêneo possível nos cursos de graduação. Uma das propostas para alcançar esse objetivo é a utilização de ferramentas gráficas de apoio ao ensino [Soares, 2004].

O uso de ferramentas gráficas educacionais pode melhorar o aprendizado do aluno, pois a rápida resposta dada por estas ferramentas [Araújo, 2002]: i) o encoraja a se autocorriger; ii) o leva à experimentação e à descoberta do conhecimento; e iii) promove a imediata discussão dos erros e acertos, conduzindo-o a uma melhor compreensão do assunto abordado.

Uma disciplina dos cursos de computação e áreas afins que tem recebido constantes esforços, com o intuito de melhorar as estratégias de ensino, bem como para o desenvolvimento de ferramentas gráficas educacionais, é Algoritmos e Estrutura de Dados (AED) [Black, 2013; Santos e Costa, 2005; Flávio, 2013; Azul e Mendes, 1998]. Contudo, ainda há carência de estratégias e ferramentas disponíveis na literatura para ensino dessa disciplina, cujo enfoque está sob a análise da eficiência dos algoritmos. Neste artigo, o conceito de eficiência está relacionado ao número de comparações e/ou de trocas realizadas por um determinado algoritmo. A apresentação dos conceitos relacionados à análise da eficiência de algoritmos em períodos iniciais dos cursos de graduação em computação não é muito comum, porém a introdução tardia destes conceitos é um problema, pois, segundo Ginat (1996): i) pode levar os alunos a desenvolverem algoritmos ineficientes, que teriam impactos negativos quando utilizados na prática, principalmente em sistemas com poucos recursos computacionais (por exemplo, dispositivos embarcados); e ii) após estabelecerem mentalmente um modelo de programação, os alunos tendem a ter maior dificuldade em mudar seus estilos de programação posteriormente. Outra limitação encontrada na maioria das ferramentas para ensino de AED disponíveis na literatura é a baixa extensibilidade dessas ferramentas, ou seja, não há mecanismos que facilitem a extensão de sua funcionalidade, permitindo que novos tipos de algoritmos sejam adicionados a ela. Permitir a adição de novas funcionalidades é interessante, pois torna a ferramenta adaptável às necessidades de seus usuários.

Dessa forma, o objetivo deste trabalho é apresentar uma ferramenta educacional gráfica extensível para ensino de métodos de ordenação e pesquisa, com ênfase na análise da eficiência de algoritmos, denominada AESDA. A ideia é que esta ferramenta permita ao usuário (professor/aluno): i) criar através de uma interface gráfica, conjuntos de dados numéricos, cujos valores podem ser escolhidos pelo usuário ou aleatoriamente; ii) observar graficamente e de forma animada a execução dos métodos de ordenação *Selection Sort*, *Insertion Sort*, *Bubble Sort* e Busca Sequencial e Binária (tópicos contemplados nas disciplinas AED) sobre os conjuntos de dados criados; e iii) analisar com o auxílio de gráficos a eficiência destes algoritmos de forma independente ou comparando-os com outros algoritmos da mesma categoria. Além disso, a ferramenta proposta visa a ser flexível, no que tange à introdução de novas funcionalidades. Sendo assim, o usuário poderá desenvolver seus próprios algoritmos e observar a execução dos mesmos de forma gráfica e com recursos de animação, utilizando a estrutura de classes disponível na AESDA, sem a necessidade de conhecer detalhes da API de animação utilizada no desenvolvimento.

Este artigo encontra-se organizado da seguinte forma: na Seção 2 são comentadas, sucintamente, algumas características do ensino de Algoritmos e Estruturas de Dados, destacando-se algumas de suas principais dificuldades e desafios. Na Seção 3 são discutidos alguns trabalhos relacionados. Na Seção 4 é apresentada a ferramenta AESDA, dando ênfase sobre os detalhes de sua arquitetura, interface e funcionamento.

Na seção 5 são apresentados os resultados obtidos a partir da avaliação da ferramenta AESDA, realizada com graduandos em ciência da computação. Por fim, na Seção 6 estão as considerações finais e trabalhos futuros.

2. Ensino de Algoritmos e Estrutura de Dados

O ensino da disciplina de Algoritmos e Estrutura de Dados possui alguns complicadores, pois se trata de uma disciplina cujos conteúdos exigem dos alunos capacidades, como abstração, generalização, transferência de conceitos, dentre outras, as quais muitas vezes não foram desenvolvidas durante sua vida escolar [Gomes e Mendes, 2007]. Além disso, tais disciplinas são ministradas seguindo uma progressão encadeada por fatores lógicos e didáticos dos conteúdos. Isto é, a transmissão do conteúdo é realizada em etapas interdependentes e a assimilação do conhecimento após cada etapa é importante, pois cada uma dessas é pré-requisito para o entendimento da próxima [Gomes da Costa, 2011].

Neste contexto, qualquer motivo que leve à desconcentração ou desmotivação do aluno pode ter consequências sérias para o aprendizado dos conceitos. Segundo Soares (2004), ferramentas educacionais gráficas podem ser utilizadas com o intuito de amenizar tais problemas de ensino. Uma ferramenta educacional é um conjunto de recursos computacionais projetados com a intenção de serem usados em contexto de aprendizagem [Cano, 1998]. Segundo Mercado (2002), tais ferramentas podem facilitar a percepção dos conceitos da disciplina por parte do aluno, trazendo para o monitor do computador significativa quantidade de estímulos (como imagens, textos e animações) que eliminam as tarefas que exigem exercício de processos repetitivos e maçantes. Assim, o aluno é estimulado a dar maior atenção ao “Por quê?”, “Como funciona?”, “O que é?” de conceitos e de processos de maneira projetada visando à facilidade de aprendizado.

Entretanto, segundo Coscarelli (1998), para que uma ferramenta possa contribuir efetivamente para o processo de ensino-aprendizagem, ela deve: i) propiciar suporte para a reflexão; ii) estimular e criar oportunidades para que o aluno pense em ideias sob várias perspectivas; iii) fornecer *feedback* rico e explicativo; iv) explorar erros como oportunidades para desenvolver o aprendizado; v) explorar diferenças individuais de conhecimento e habilidades; e vi) fornecer medidas significativas de avaliação, por exemplo, um relatório do uso de uma instância da ferramenta.

3. Trabalhos Relacionados

Nesta seção, são descritas sucintamente as seguintes ferramentas: i) TBC/AED - Treinamento Baseado em Computador para Algoritmos e Estruturas de Dados [Santos e Costa, 2005]; ii) ASTRAL - *Animation of Data Structures and Algorithms* [Garcia *et al*, 1997]; e iii) ODIN - Ambiente *Web* de apoio ao Ensino de Estruturas de Dados Lista Encadeada [Madeira, 2005].

a) **TBC/AED:** é uma ferramenta gráfica desenvolvida para *web* que possui as seguintes características: i) possui *links* explicativos para os algoritmos, evitando a necessidade de aprendizado via tutorial ou manual; ii) apresenta o conteúdo teórico de forma simples, com o intuito de familiarizar o aluno com o assunto; iii) apresenta processo de execução dos algoritmos de forma gráfica e passo a passo, o que pode

melhorar a visualização e o entendimento; e v) possui legendas explicativas, que ilustram as etapas do processo de apresentação de algoritmos.

b) ASTRAL: é um ambiente de programação para produção de animações de algoritmos e de estruturas de dados com propósito instrucional. Essa ferramenta conta com a representação de uma grande variedade de algoritmos, como árvores, grafos, algoritmos de ordenação, gramáticas entre outros. O usuário deve fazer o *download* do tipo de algoritmo que deseja executar em seu computador. Essa ferramenta também permite que ele calcule a quantidade de execuções e o tempo que determinado algoritmo levou para realizar seu trabalho.

c) ODIN: é um ambiente de apoio ao ensino de estrutura de dados que simula, especificamente, o funcionamento da lista encadeada. O módulo do ODIN responsável pela parte de apresentação gráfica das funções primitivas de lista encadeada é composto de três partes: i) painel onde é apresentado o código fonte de cada função da lista; ii) painel onde são visualizados os nós da estrutura de dados representada; e iii) botões que possibilitam a interação do usuário com o ambiente.

A ferramenta proposta neste trabalho, AESDA, engloba muitas das funcionalidades descritas anteriormente, contudo, apresenta as seguintes diferenças: i) na ferramenta AESDA, o aluno tem acesso ao seu código e pode verificar qual parte do mesmo está sendo executada ao mesmo tempo em que ele visualiza a execução animada das estruturas de dados; ii) leva em consideração a comparação da eficiência dos algoritmos exibindo os resultados de forma gráfica (a ferramenta ASTRAL apresenta apenas quantidade de comparações e o tempo de execução, mas não mostra nenhum tipo de gráfico ao usuário); e iii) permitir que o usuário incremente sua funcionalidade de acordo com suas necessidades, sem precisar conhecer profundamente detalhes da tecnologia utilizada no desenvolvimento da AESDA. A Tabela 1 resume a comparação das ferramentas para ensino de AED descritas anteriormente. Como pode ser observado, a maior parte das ferramentas analisadas possuem recursos gráficos e de animação, além de estarem disponíveis em ambiente *web*. Porém, as características de comparação da eficiência dos algoritmos, extensão da funcionalidade da ferramenta e visualização animada de trechos de código só aparecem em uma pequena parcela das ferramentas.

Tabela 1 - Comparação entre ferramentas para ensino de AED.

Critérios	AESDA	TBC/AED	ODI N	ASTRAL
Possui recursos gráficos e de animação.	x	x	x	x
Permite a comparação da eficiência dos algoritmos disponíveis na ferramenta.	x			x
Pode ser executada na <i>web</i> .	x	x	x	
Permite extensão fácil da funcionalidade da ferramenta.	x			
Propicia a visualização animada do trecho de código do algoritmo que está sendo executado.	x			

4. Ferramenta AESDA

Para o desenvolvimento da AESDA, buscou-se atender aos requisitos para uma boa ferramenta educacional, citados por Coscarelli (1998) (Seção 2). À medida que a ferramenta AESDA for apresentada neste artigo, será discutido como tais requisitos foram contemplados. Devido a limitações de espaço, detalhes sobre o funcionamento de alguns recursos da AESDA foram omitidos e podem ser visualizados em Silva (2012).

Na Figura 1 é apresentada a arquitetura da ferramenta AESDA, que é dividida em: i) Módulo de Interface Gráfica; ii) Módulo Gráfico; e iii) Módulo Algorítmico.

4.1 Módulo de Interface Gráfica

No **Módulo de Interface Gráfica**, o usuário pode escolher um dos algoritmos existentes na ferramenta, e então executá-lo passo a passo. Isso significa que o usuário pode visualizar graficamente e com recursos de animação, o comportamento de cada trecho do código do algoritmo escolhido. A Figura 2 apresenta a tela inicial da AESDA, na qual se pode observar um *menu* interativo subdividido em:

- i) **executar algoritmos.** Ao clicar neste item, um novo *menu* será apresentado no qual o usuário poderá escolher o algoritmo que deseja observar em execução passo a passo. A AESDA disponibiliza algoritmos para os métodos de ordenação *Selection Sort*, *Insertion Sort*, *Bubble Sort* e de busca *Binária* e *Sequencial*; e
- ii) **comparar algoritmos.** Ao escolher esta opção o usuário poderá escolher um ou mais algoritmos e observar através de um gráfico a eficiência destes em termos de número de trocas e de comparações executadas.

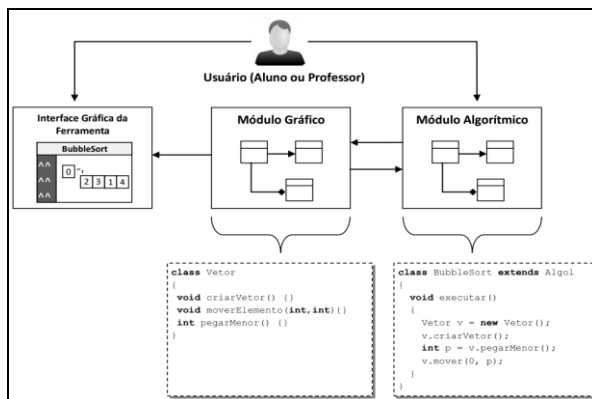


Figura 1 - Arquitetura da AESDA.

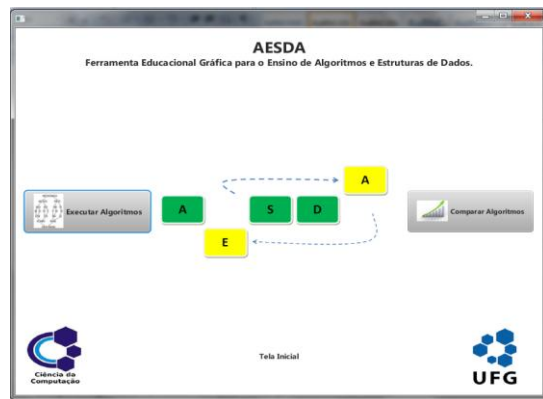


Figura 2 - Tela inicial da AESDA.

A Figura 3 e Figura 4 apresentam as telas de execução do método de ordenação *SelectionSort* e de comparação da eficiência de dois algoritmos, respectivamente. Na Figura 3 (1) há um painel, no qual é apresentado: i) o código do algoritmo na aba “Código Fonte”; ii) um texto explicativo sobre o funcionamento do algoritmo em “Informações”; e iii) a legenda das cores presentes na animação na aba “Legenda”. A Figura 3 (2) contém o painel de animação, no qual ocorrerá a execução gráfica e animada do algoritmo em questão. Os elementos existentes nesse painel mudarão de cor e de posição de acordo com a parte do código que está sendo executada. Além disso, mensagens autoexplicativas sobre o que está sendo executado no código aparecem na tela e são gravadas no componente “Console”, localizado logo abaixo dos elementos de animação.

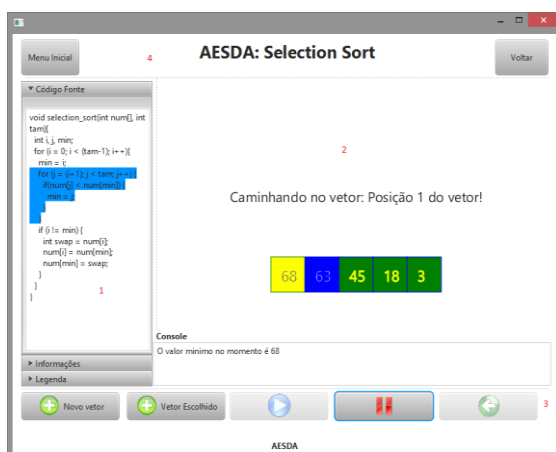


Figura 3 - Tela de execução de um método de ordenação.

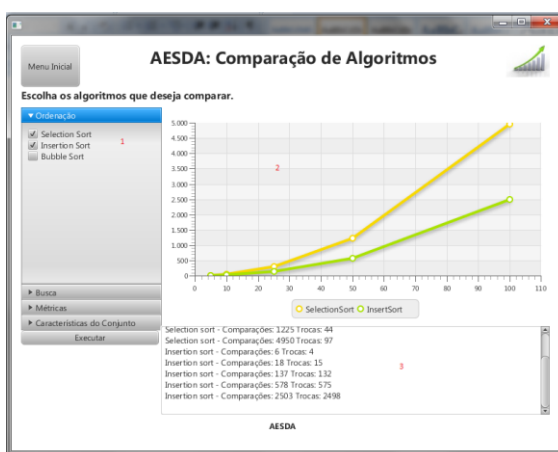


Figura 4 - Tela de comparação da eficiência de algoritmos.

Na Figura 3 (3) está o *menu* de execução do algoritmo, nesse caso, há os seguintes botões: i) **Play**, **Pause** e **Retornar ao Início** (representados por símbolos), que servem, respectivamente, para iniciar a execução do algoritmo, pausar a execução em determinado momento e retornar a execução ao início; ii) **Vetor Escolhido**, que permite ao usuário criar um vetor com valores escolhidos por ele; e iii) **Novo Vetor**, que permite criar um vetor com valores aleatórios. Na Figura 3 (4), estão os botões **Voltar** e **Menu Inicial** que retornam a tela anterior e a tela inicial, respectivamente.

Os critérios 1 (propiciar suporte para a reflexão) e 3 (fornecer *feedback* rico e explicativo) propostos por Coscarelli (Seção 2), são contemplados na AESDA por meio dos recursos de animações de vetores, da marcação do trecho de código correspondente à animação que está ocorrendo e das mensagens autoexplicativas enviadas ao usuário durante a execução do algoritmo. Quanto aos critérios 2 (estimular e criar oportunidades para que o aluno pense em ideias sob várias perspectivas) e 4 (explorar erros como oportunidades para desenvolver o aprendizado), a ferramenta AESDA contempla-os ao permitir que o usuário possa criar vetores de elementos da maneira como achar conveniente. Isso permite que o aluno realize testes sobre vários tipos de vetores de entrada, o que pode melhorar sua compreensão sobre os conceitos tratados pela ferramenta.

A Figura 4 apresenta a tela de comparação de algoritmos que possui: i) uma área para apresentação do gráfico comparativo da eficiência dos algoritmos (Figura 4 - 2); ii) uma listagem com informações sobre quantidade de comparações e trocas realizadas (Figura 4 - 3); e iii) uma área para seleção dos algoritmos a serem comparados, as métricas a serem utilizadas (quantidade de comparações ou trocas) e as características dos conjuntos de entrada (vetores aleatórios, crescentes ou decrescentes) (Figura 4 - 1).

Essa funcionalidade da AESDA pode auxiliar também na concretização do segundo critério proposto por Coscarelli, pois ao visualizar a comparação entre dois algoritmos o aluno poderá refletir e se sentir estimulado a criar algoritmos mais eficientes.

4.2 Módulo Gráfico

Continuando com a explicação da arquitetura da AESDA (Figura 1), o **Módulo Gráfico** consiste em um conjunto de classes que permitem a criação das estruturas gráficas utilizadas na ferramenta, bem como a animação dos elementos destas estruturas. Por exemplo, na Figura 1, observa-se a classe `Vetor`, que possui métodos responsáveis pela: i) criação de uma estrutura gráfica “vetor” com elementos aleatórios (`criarVetor`); ii) obtenção do índice de um elemento desta estrutura e marcá-lo com uma cor diferente (`marcarElemento`); e iii) movimentação de um elemento desta estrutura de uma posição para outra (`moverElemento`). É por meio deste conjunto de classes que o usuário poderá estender as funcionalidades da AESDA, incorporando novos tipos de algoritmos. Por exemplo, supondo que o professor da disciplina AED queira ensinar o algoritmo de Busca Ternária aos seus alunos utilizando a ferramenta AESDA, ele poderá estender sua funcionalidade, utilizando como apoio as classes do Módulo Gráfico. A vantagem é que os usuários poderão utilizar recursos gráficos e de animação em seus algoritmos, sem a necessidade de conhecimento prévio das APIs ou *frameworks* para criação de gráficos e animação.

4.3 Módulo Algorítmico

O **Módulo Algorítmico** consiste de um conjunto de classes que representam os algoritmos disponíveis na ferramenta AESDA. Uma classe deste módulo pode utilizar as classes do Módulo Gráfico para a execução gráfica de um determinado algoritmo. Por exemplo, ainda segundo a Figura 1, a classe `BubbleSort` é responsável pela execução desse algoritmo. A existência desta classe indica que uma opção *BubbleSort* estará disponível no menu de algoritmo da AESDA. Quando o usuário selecionar o item no *menu*, o método `executar` da classe `BubbleSort` será executado. No código deste método encontra-se a lógica de execução do algoritmo de ordenação *BubbleSort*.

Para implementação da ferramenta *AESDA* foram utilizadas as seguintes tecnologias: i) linguagem de programação Java; e ii) *framework* para desenvolvimento de aplicações gráficas, JavaFX. Tais tecnologias foram escolhidas, pois tem alta portabilidade, as aplicações desenvolvidas com elas podem ser disponibilizadas via *web* e têm sido constantemente atualizadas.

5. Avaliação da Ferramenta AESDA

Para avaliação da ferramenta AESDA foi utilizado o modelo de aceitação de tecnologia (TAM - *Technology Acceptance Model*) [Davis *et al.*, 1989]. Esse modelo visa explicar o comportamento das pessoas no que diz respeito à aceitação de uma tecnologia. Segundo Davis *et al.* (1989), o modelo TAM define dois constructos básicos: i) utilidade percebida, que mede o quanto uma pessoa acredita que usar determinada tecnologia aumenta seu desempenho no trabalho; e ii) facilidade de uso percebida, que mede o quanto uma pessoa acredita que o uso de determinada tecnologia é fácil.

Para isso, o modelo TAM sugere a criação de questionários, aos quais, são atribuídas afirmações relacionadas à facilidade e utilidade de uso da tecnologia em análise. Para cada afirmação, o respondente poderá escolher uma dentre as seguintes opções “Discordo Totalmente”, “Discordo Fortemente”, “Discordo Parcialmente”, “Neutro”, “Concordo Parcialmente”, “Concordo Fortemente” e “Concordo Totalmente”, conforme sua opinião. Um usuário, após ter passado por um treinamento sobre a

AESDA e ter utilizado-a, pode mensurar qualitativamente seu sentimento de utilidade e facilidade de uso, com base nas alternativas citadas anteriormente.

O questionário desenvolvido possui dezenove afirmações (nove relacionadas à facilidade de uso e dez relacionadas à utilidade percebida) e a avaliação da AESDA foi realizada com doze graduandos em Ciência da Computação da Universidade Federal de Goiás (*Câmpus Jataí*). As afirmações do questionário, bem como os resultados da avaliação realizada são apresentadas na Tabela 2 e nos Gráficos 1, 2 e 3, respectivamente.

Tabela 2 – Afirmações utilizadas no questionário avaliativo.

Afirmações relacionadas à facilidade de uso		Afirmações relacionadas à utilidade percebida	
1	Eu gosto de trabalhar com a AESDA.	10	Utilizar a AESDA é importante e adiciona valor ao meu trabalho/estudo.
2	O acesso à AESDA é simples.	11	A AESDA é útil no processo de ensino/aprendizado dos conceitos de algoritmos de ordenação e busca.
3	Usar a AESDA é uma boa ideia.	12	Usar a AESDA pode aumentar meu desempenho durante os estudos ou ensino da disciplina Estrutura de Dados.
4	Na AESDA eu sempre sei onde estou e como chegar aonde quero chegar.	13	A AESDA pode facilitar a realização do meu estudo/trabalho.
5	Os recursos de navegação da AESDA são todos claros e fáceis de achar.	14	A AESDA produz os resultados que espero de uma ferramenta de suporte ao ensino de Estrutura de Dados.
6	Minha interação com a AESDA é clara e compreensível.	15	Eu tenho a intenção de utilizar a AESDA ao longo dos próximos semestres.
7	Na AESDA é fácil encontrar a informação que desejo.	16	Eu pretendo integrar a AESDA à minha rotina de trabalho/estudo.
8	A AESDA possui visual/interface atraente.	17	Eu recomendarei o uso da AESDA.
9	Mesmo antes de clicar em um botão na AESDA eu já sei a ação dele.	18	Os algoritmos de ordenação e busca foram abordados por completo.
		19	Todos os algoritmos de ordenação e busca presentes na AESDA foram abordados de forma correta.

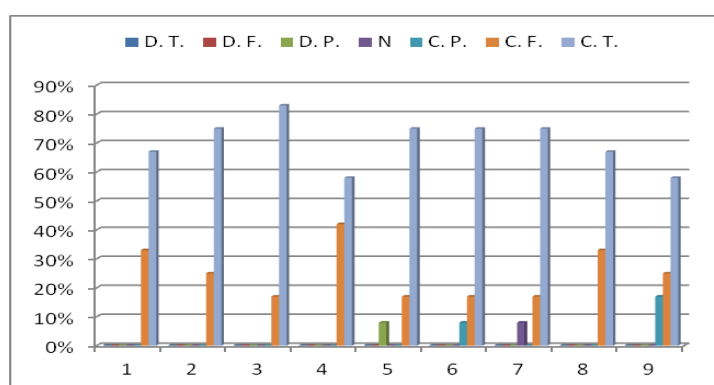


Gráfico 1: Resultados para o constructo “Facilidade de Uso”.

Com base no Gráfico 1, pode-se perceber que para as questões de 1 até 9, a maioria dos usuários escolheram a opção 7 que significa “Concordo Totalmente” (entre 65% e pouco mais de 80% das vezes). As demais opções, inferiores a 6 (Concordo Fortemente), foram escolhidas em menos de 10% dos resultados, além disso em apenas uma afirmação houve a ocorrência de escolhas abaixo da opção “Neutro”. A afirmação que recebeu a pior nota, com relação ao constructo “Facilidade de Uso”, foi a cinco: “Os

recursos de navegação da AESDA são todos claros e fáceis de achar”. Esse é um resultado importante e que deverá ser analisado em trabalhos futuros.

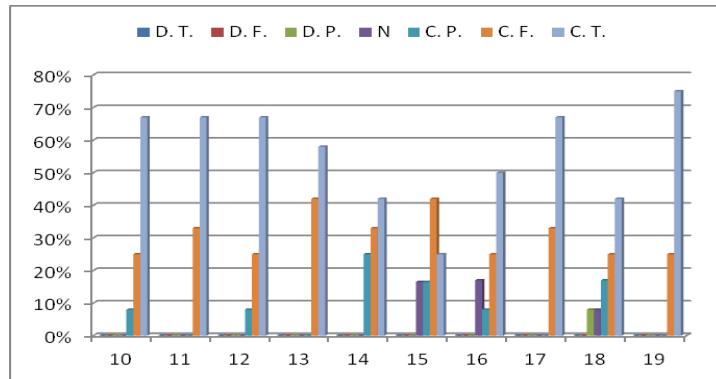


Gráfico 2: Resultados para o Constructo “Utilidade de uso”.

Já no Gráfico 2, que diz respeito ao construto “Utilidade Percebida” das 10 afirmações feitas, apenas para a de número 15 (“Eu tenho a intenção de utilizar a AESDA ao longo dos próximos semestres”) a alternativa “Concordo Totalmente” não foi a mais escolhida. Acredita-se que esse resultado se deu, pois todos os usuários participantes da avaliação já haviam cursado ou estavam finalizando a disciplina de Estrutura de Dados. Em contrapartida, para a afirmação 17, que está relacionada ao fato de os usuários indicarem o uso da ferramenta, a maioria das notas dadas pelos usuários foi “Concordo Totalmente”. O que pode indicar que, apesar de não terem a intenção de utilizar tal ferramenta nos próximos semestres, os usuários acharam a ferramenta útil a ponto de indicá-la a outros colegas.

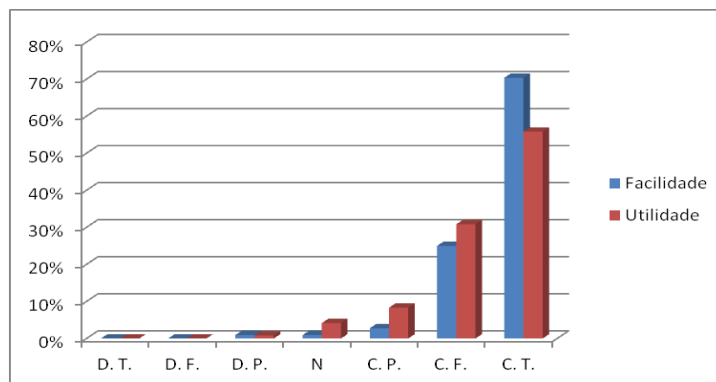


Gráfico 3: Resumo dos resultados da avaliação.

Por fim, o Gráfico 3 apresenta a porcentagem de votos para cada uma das alternativas possíveis de serem escolhidas pelos usuários, com base nos constructos “Facilidade de Uso” e “Utilidade Percebida”. O Gráfico 3 mostra que, respectivamente, aproximadamente 70% e 25% das respostas para o constructo “Facilidade de Uso” foram “Concordo Totalmente” e “Concordo Fortemente”. Já para o constructo “Utilidade de Uso” as mesmas respostas foram escolhidas cerca de 55% e 30% das vezes, respectivamente. Esses resultados reforçam a necessidade de uma análise mais detalhada da usabilidade da ferramenta AESDA.

6. Considerações Finais

Neste trabalho apresentou-se a ferramenta educacional gráfica extensível de apoio ao ensino de algoritmos de Ordenação e Busca, denominada AESDA. Também foi apresentada uma breve avaliação da AESDA com graduandos em ciência da computação, na qual observou-se bons resultados com relação à facilidade de uso e utilidade dessa ferramenta por parte de seus usuários.

Como trabalhos futuros, tem-se a intenção de inserir novos algoritmos na AESDA, dentre eles estão: Pilhas, Filas, Listas, Árvores, entre outros. O intuito da criação de mais algoritmos é tornar a ferramenta mais completa, abrangendo a maior parte do conteúdo da disciplina de AED. Tem-se ainda a intenção de criar mecanismos para que o usuário possa inserir seu próprio algoritmo e logo em seguida visualizar as animações criadas a partir dele. Com o *framework* gráfico criado nesta ferramenta já é possível a criação, porém o usuário precisa modificar uma parte do código fonte da ferramenta. Por fim, devem ser realizadas avaliações mais robustas da ferramenta AESDA, principalmente com relação à sua usabilidade. O objetivo é detectar seus principais problemas de falta de usabilidade e corrigi-los em uma próxima versão da ferramenta.

Referências

- Araújo, C. R. “Uma proposta de ferramenta de apoio à educação musical via Web usando Java XML”. Dissertação de Mestrado. FEEC/UNICAMP. 2002.
- Azul, A. A.; Mendes, A. J. “EDDL: Um Programa Didático sobre Estruturas de Dados Dinâmicas Lineares”. 3º Simpósio Investigação e Desenvolvimento de Software Educativo. Évora. Portugal. 1998.
- Black, P. E. “Dictionary of Algorithms and Data Structures”. National Institute of Standards and Technology. Disponível em <http://www.nist.gov/dads/>. Acesso em: Abril 2013.
- Cano, C. A. “Os Recursos da Informática e os Contextos de Ensino e Aprendizagem”. ArtMed, 1998.
- Coscarelli, C. V. O. “Uso da Informática como instrumento de Ensino e Aprendizagem”, Revista Presença Pedagógica, Editora Dimensão. Disponível em: <<http://www.presencapedagogica.com.br/capa6/artigos/20.pdf>>. Acesso em: Abril de 2013.
- Davis, F.D.; Bagozzi, R. P.; Warshaw P.R. “User Acceptance of Computer Technology: A Comparison of two Theoretical Models. Management Science. v. 35, n. 8, p. 982-1003, 1989.
- Flávio, D. TED – “Tutorial de Estruturas de Dados”. Disponível em: <http://www.tutorialdeestruturadedados.8m.com>. Acesso: Abril de 2013.
- Ginat, David, “Efficiency of algorithms for programming beginners”, Technical Symposium on Computer science education. Volume 28 Issue 1. Março, 1996.
- Gomes, A.; Mendes, A. J. “Learning to program - difficulties and solutions. International Conference on Engineering Education”. ICEE 2007, Setembro 2007.
- Gomes da Costa, F. H. “Objeto de Aprendizagem para o ensino de Estruturas de Dados”. Monografia de Graduação. UnB, 2011.
- Madeira, M. F., Simões, P. W. T. de A., e Martins, P. J. “ODIN - ambiente web de apoio ao ensino de estruturas de dados lista encadeada”. 1º Congresso Sul catarinense de Computação, 2005.
- Mercado, L. P. L. “Novas Tecnologias na Educação: Reflexões Sobre a Prática”. EDUFAL, 2002.
- Santos, R. P. e Costa, H. A. X. TBC-AED: “Um Software Gráfico para Apresentação de Algoritmos e Estruturas de Dados aos Iniciantes em Computação e Informática”. I Congresso de Computação do Sul do Mato Grosso - COMPSULMT. Rondonópolis, MT, Brasil, 2005.
- Silva, M. “AESDA – Ferramenta Educacional Gráfica para Ensino de Algoritmos de Ordenação e Pesquisa com Ênfase na Análise da Eficiência dos Algoritmos”. Monografia de Graduação. Ciência da Computação da UFG/Câmpus Jataí, 2012.
- Soares, T. C. A. P., Cordeiro E. S., Stefani Í. G. A., Tirelo, F. “Uma Proposta Metodológica para o Aprendizado de Algoritmos em Grafos Via Animação Não-Intrusiva de Algoritmos”. WEIMIG 2004