

Universidade Federal de Ouro Preto - UFOP
Instituto de Ciências Exatas e Biológicas - ICEB
Departamento de Computação - DECOM

$(PAA)^2$

PACOTE DE APOIO A APRENDIZAGEM DE
PROJETO E ANÁLISE DE ALGORITMOS:
IMPLEMENTAÇÃO EM JAVA

Aluno: Thiago Andrade Pereira
Matricula: 07.1.4177

Orientadora: Andréa Iabrudi Tavares

Ouro Preto
2 de julho de 2011

Universidade Federal de Ouro Preto - UFOP
Instituto de Ciências Exatas e Biológicas - ICEB
Departamento de Computação - DECOM

$(PAA)^2$

PACOTE DE APOIO A APRENDIZAGEM DE PROJETO
E ANÁLISE DE ALGORITMOS: IMPLEMENTAÇÃO EM
JAVA

Relatório de atividades desenvolvidas apresentado ao curso de Bacharelado em Ciência da Computação, Universidade Federal de Ouro Preto, como requisito parcial para a conclusão da disciplina Monografia I (BCC390).

Aluno: Thiago Andrade Pereira
Matricula: 07.4177

Orientadora: Andréa Iabrudi Tavares

Ouro Preto
2 de julho de 2011

Resumo

O Projeto e Análise de Algoritmos é um campo da ciência da computação que visa introduzir conceitos mais avançados de desenvolvimento de algoritmos, análise de complexidade de problemas computacionais, comparar diferentes algoritmos para a solução de um mesmo problema, descrever e empregar os princípios, métodos e técnicas fundamentais para projeto de algoritmos corretos e eficientes. Por essa razão, o estudo de projeto e análise de algoritmos é fundamental para toda a computação. Este trabalho apresenta uma biblioteca de funções que implementam as principais técnicas de projeto de algoritmos para auxiliar no ensino da disciplina de Projeto e Análise de Algoritmos ministrada para o curso de Ciência da Computação da Universidade Federal de Ouro Preto.

Palavras-chave: Algoritmos. Projeto. Análise.

Sumário

1	Introdução	1
2	Justificativa	2
3	Objetivos	3
3.1	Objetivo geral	3
3.2	Objetivos específicos	3
4	Metodologia	4
5	Desenvolvimento	5
5.1	Escopo do Projeto	5
5.2	Ferramentas Utilizadas	5
5.3	Status do Projeto e Primeiros Resultados	5
6	Trabalhos Futuros	6
6.1	Teste da Caixa Preta	6
7	Cronograma de atividades	7

Lista de Tabelas

1	Cronograma de Atividades.	7
---	-----------------------------------	---

1 Introdução

Projeto e Análise de Algoritmos é a disciplina da Ciência da Computação que estuda o comportamento dos algoritmos, e que tem papel decisivo no projeto de algoritmos eficientes. O propósito principal desse curso é analisar e projetar algoritmos eficientes, através do estudo de uma variedade de algoritmos já bem conhecidos e considerados clássicos na Computação [10]. Este trabalho propõe a criação de uma biblioteca que implementa um conjunto de paradigmas de projeto de algoritmos estudados na disciplina, que servirá como ferramenta auxiliar no ensino de Projeto e Análise de Algoritmos (BCC241), disciplina ministrada para o curso de Ciência da Computação na Universidade Federal de Ouro Preto. Espera-se que o uso dessa biblioteca como recurso didático contribuirá na consolidação do conteúdo pelos alunos.

Esse projeto toma como base a aprendizagem baseada em exemplos [2]. Através de uma série de exemplos de um conceito, o aluno primeiramente analisa esses casos específicos para depois chegar à compreensão dos princípios e teorias. Para cada paradigma de projeto de algoritmos, teremos a implementação da mesma e pelo menos um estudo de caso (aplicação). Por trazer implementada soluções, diferentes exemplos poderão ser apresentados aos alunos, em diferentes níveis de complexidade e completeza, o que permitirá que o foco de aprendizagem seja direcionado pelo professor.

Conceitos de orientação a objetos e programação genérica [5] serão fortemente empregados nesse trabalho, no desenvolvimento da biblioteca. A programação genérica nos permite criar modelos gerais, ou seja, métodos genéricos e classes genéricas, permitindo aos programadores que especifiquem com uma única declaração de método, um conjunto de métodos relacionados ou, com uma única declaração de classe, um conjunto de tipos relacionados. As classes abstratas e métodos abstratos serão parte da arquitetura da biblioteca, o que contribuirá para a experiência do aluno em programação de forma genérica e orientação à objetos.

2 Justificativa

A disciplina de Projeto e Análise de Algoritmos é fundamental e obrigatória para o curso de Ciência da Computação. Uma biblioteca que implementa os paradigmas de projeto de algoritmos beneficiaria o aluno na aprendizagem do conteúdo, já que proporciona agilidade e facilidade de utilização do paradigma para resolver um problema. Além disso, a partir de uma solução genérica, aspectos específicos para o melhor desempenho de um paradigma – por exemplo, diferentes estratégias de exploração do *Branch-and-bound* – podem ser investigados e implementados pelos alunos com maior facilidade. O objetivo é que o aluno entenda onde, quando e o porquê de cada etapa do paradigma para a solução do problema, por isso a implementação da técnica está contida na biblioteca, cabe ao aluno especificar os passos para um problema específico.

Além disso, é comum os alunos terem dificuldades de implementação e o uso desse pacote de funções proporcionaria familiaridade com boas práticas de programação e aprimoraria conceitos de orientação a objetos e programação genérica.

3 Objetivos

3.1 Objetivo geral

O objetivo deste trabalho é implementar uma biblioteca com alguns dos principais paradigmas de projeto de algoritmos para ser usada na disciplina de Projeto e Análise de Algoritmos. Os paradigmas implementados serão: Dividir para Conquistar, Programação Dinâmica, Abordagem Gulosa, *Backtracking* e *Branch-and-Bound*. Todas elas serão implementadas de forma genérica para proporcionar a solução quaisquer natureza de problema em que é viável a aplicação da técnica.

3.2 Objetivos específicos

- Gerar uma documentação de código para ser usada como exemplo de documentação.
- Gerar uma página do projeto na *web* para *download* do código.
- Realizar testes de desempenho em algumas técnicas.
- Exemplificar boas técnicas de documentação.
- Apresentar instrumentação de código para tomada de tempo e análise empírica comparativa.

4 Metodologia

Para a implementação da biblioteca será utilizada a linguagem de programação Java [8] e o ambiente de programação *Eclipse IDE for Java Developers* na versão *Helios Service Release 1* [7]. Para a documentação do código será utilizada a ferramenta *Doxygen* [6], que é um sistema de documentação de códigos para diversas linguagens de programação.

A metodologia adotada para o desenvolvimento do trabalho será composta por etapas. Para cada uma das cinco técnicas de projetos a ser implementada será realizado:

- O projeto das classes e dos algoritmos;
- Implementação;
- Documentação;
- Testes;

Para o auxílio nos projetos de classe, algoritmos e implementação, será seguida a bibliografia dos autores de [1], [11] e [3].

5 Desenvolvimento

Nesta seção, serão expostas as atividades realizadas durante os primeiros meses do projeto.

5.1 Escopo do Projeto

A princípio foi feito a definição do escopo do projeto, ou seja, o que se pretende realizar estabelecendo seu raio de ação e definindo seus limites. Obtivemos como escopo do projeto os seguintes elementos:

- **Definição do problema/situação geradora do projeto:** Apresentado na Seção 1.
- **Justificativa:** Apresentado na Seção 2.
- **Objetivos gerais e específicos:** Apresentado na Seção 3.
- **Resultados esperados:** Uma biblioteca java com a implementação das cinco técnicas de projeto de algoritmos estudadas na disciplina e uma página web que disponibilizará o *download* da biblioteca e documentações.
- **Abrangência do projeto:** O público alvo do produto resultante do projeto são os alunos da disciplina de Projeto e Análise de Algoritmos.

5.2 Ferramentas Utilizadas

Com o entendimento do que precisa ser feito foi realizado um estudo em cima das ferramentas a serem utilizadas durante o projeto.

Para a ferramenta de desenvolvimento Eclipse foi realizado um estudo na sua documentação *Workbench User Guide* [12] com o objetivo de identificar os procedimentos de geração e instalação de bibliotecas, criação e gerenciamento de pacotes e importação de projetos.

Na ferramenta de documentação Doxygen foi feito um estudo em seu manual disponibilizado em [6], com o objetivo de identificar os procedimentos de instalação, integração com a ferramenta Eclipse e estilos de documentações de códigos.

5.3 Status do Projeto e Primeiros Resultados

Até o momento foi realizado um estudo bibliográfico da técnica de Dividir para Conquistar baseado nos autores [11] e [3], e um estudo da técnica Backtracking baseado no autor [11]. Como resultado obtivemos um protótipo da biblioteca com a implementação dos algoritmos MergeSort e QuickSort utilizando a técnica de Dividir para Conquistar.

Foram gerados tutoriais de apoio para os usuários da biblioteca contendo os passos necessários para instalação da biblioteca e utilização dos algoritmos exemplos MergeSort e QuickSort.

Foi disponibilizado as primeiras documentações utilizando a ferramenta Doxygen e por fim foi desenvolvido a página web contendo as documentações de códigos, protótipo da biblioteca e tutoriais de apoio para os usuários, (disponível em <http://www.decom.ufop.br/paapaa>).

6 Trabalhos Futuros

Dando continuidade ao projeto, os próximos passos serão desenvolvidos na disciplina subsequente denominada Monografia II(BCC391). Os passos serão:

- Implementação da técnica *Backtracking* e documentação.
- Revisão bibliográfica da técnica *Branch-and-Bound*, implementação da técnica e documentação.
- Revisão bibliográfica da técnica Programação Dinâmica, implementação da técnica e documentação.
- Revisão bibliográfica da técnica Abordagem Gulosa, implementação da técnica e documentação.
- Teste da Caixa Preta.
- Redigir a Monografia.

6.1 Teste da Caixa Preta

Segundo [9], o teste da caixa preta, também chamado de teste comportamental tenta encontrar erros das seguintes categorias: (i) funções incorretas ou omitidas; (ii) erros de interface; (iii) erros de estrutura de dados ou de acesso à base de dados externa; (iv) erros de comportamento ou desempenho e (v) erros de iniciação e término.

Faremos então o teste de Aceitação, que de acordo com [4] é um tipo de teste caixa preta, onde os próprios usuários finais, ou seja, alunos da disciplina Projeto e Análise de Algoritmos, irão validar a biblioteca verificando se esta atende aos requisitos propostos. No decorrer da disciplina, os usuários farão exercícios e trabalhos práticos utilizando as técnicas implementadas na biblioteca, e assim validando o produto ou reportando erros e inconsistências detectadas.

7 Cronograma de atividades

Na Tabela 1, é apresentado o cronograma das atividades a serem desenvolvidas para a finalização do projeto.

1. Implementação da técnica *Backtracking* e documentação;
2. Revisão bibliográfica da técnica *Branch-and-Bound*, implementação da técnica e documentação;
3. Revisão bibliográfica da técnica Programação Dinâmica, implementação da técnica e documentação;
4. Revisão bibliográfica da técnica Abordagem Gulosa, implementação da técnica e documentação;
5. Teste da Caixa Preta;
6. Redigir a Monografia;
7. Apresentação da Monografia para a banca;

Atividades	Ago	Set	Out	Nov	Dez
1	X				
2	X				
3		X			
4		X			
5	X	X	X	X	
6				X	
7					X

Tabela 1: Cronograma de Atividades.

Referências

- [1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Algoritmos - Teoria e Prática*. Elsevier, 2 edition, 2002.
- [2] Helena Noronha Cury. Estilos de aprendizagem de alunos de engenharia. *XXVIII Congresso Brasileiro de Ensino de Engenharia*, 2000.
- [3] S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani. *Algorithms*. McGraw-Hill Science/Engineering/Math, 1 edition, 2006.
- [4] Wilson de Pádua Paula Filho. Manual do engenheiro de software. Apostila, Universidade Federal de Minas Gerais, 2000.
- [5] H. M. Deitel and P. J. Deitel. *Java Como Programar*. Pearson Prentice Hall, 6 edition, 2005.
- [6] Doxygen. <http://www.stack.nl/~dimitri/doxygen/>, 1997. Visitado em 9 de junho de 2011.
- [7] Eclipse ide for java ee developers. <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/heliossr2>, 2000. Visitado em 9 de junho de 2011.
- [8] Java technology. <http://www.oracle.com/us/technologies/java/index.html/>, Visitado em 9 de junho de 2011.
- [9] Roger S. Pressman. *Engenharia de Software*. McGraw-Hill, 6 edition, 2006.
- [10] Elton José Silva. Projeto e análise de algoritmos. Apostila, Universidade Federal de Ouro Preto, Departamento de Computação, 2010.
- [11] Steven S. Skiena. *The Algorithm Design Manual*. Springer, 2 edition, 2008.
- [12] Workbench user guide. <http://help.eclipse.org/indigo/index.jsp>, Visitado em 10 de junho de 2011.