



COBENGE 2005

XXXIII - Congresso Brasileiro de Ensino de Engenharia

"Promovendo e valorizando a engenharia em um cenário de constantes mudanças"

12 a 15 de setembro - Campina Grande Pb

Promoção/Organização: ABENGE/UFPE

PROJETO MULTIDISCIPLINAR: UMA EXPERIÊNCIA PRÁTICA NO ENSINO DE PROGRAMAÇÃO EM UM CURSO DE ENGENHARIA DA COMPUTAÇÃO

Luciana A M Zaina - luciana@facens.br

Faculdade de Engenharia de Sorocaba

Rdv Senador José Ermírio de Moraes, Km 1,5

18087-125 - Sorocaba - SP

Fábio L Caversan - fcaversan@facens.br

***Resumo:** Este artigo tem como propósito apresentar uma experiência multidisciplinar realizada com um grupo de alunos de um curso de Engenharia da Computação. O trabalho foi desenvolvido pelas disciplinas de Estruturas de Dados e Algoritmos e Linguagem de Programação II. O foco foi motivar os alunos a desenvolver, através de uma linguagem orientada a objetos, soluções de problemas propostos por eles mesmos aplicando as técnicas estudadas na disciplina de Estrutura de Dados e Algoritmos.*

***Palavras-chave:** Projeto multidisciplinar, Ensino de programação, Interdisciplinaridade.*

1. INTRODUÇÃO

O ensino da programação é um tema muito estudado e discutido devido as diversas formas de se abordar o aspecto da aprendizagem nesta área. A importância do ensino da programação está não só no conteúdo abordado, mas também na capacidade de transpor os conceitos estudados para uma linguagem de programação.

GREENING (2000) destaca que no ensino da programação é fundamental adotar mecanismos que permitam ao aluno aplicar de fato o conteúdo estudado, evitando assim que a frustração possa atingir os alunos. Envolver os alunos na solução de diferentes situações problemas se constitui em uma estratégia de ensino que contribui para o desenvolvimento da aprendizagem dentro do contexto da programação.

Analisando o contexto da aprendizagem de maneira ampla deve-se destacar a importância do aluno permanecer ativo dentro do processo de ensino para que ele possa ser parte integrante e responsável do seu próprio desenvolvimento. O aluno deve encarar as aulas e seus conteúdos como uma realidade, destaca MASETTO (1998), para que ele possa desenvolver não somente o senso crítico como também se sinta estimulado a participar de discussões, estudos e pesquisas na prática.

Uma das formas de motivar os alunos a participarem do seu processo de construção do conhecimento é realizando projetos onde o aluno interage nos processos de decisão a respeito da tarefa. Outro ponto importante é o intercâmbio entre diferentes conteúdos em diferentes disciplinas.

A multidisciplinaridade se caracteriza pela intensidade das trocas e de integração entre disciplinas de um curso. A aplicação de projetos multidisciplinares em cursos de graduação da

área de computação pode se constituir de um mecanismo instigador para que o aluno possa de fato constatar como os conteúdos abordados no curso podem ser aplicados de maneira conjunta. Porém, para que um projeto multidisciplinar funcione de fato é necessário haver um procedimento de aplicação, uma coordenação que integre objetivos, atividade, procedimentos, planejamento e proporcione o intercâmbio e a troca de conhecimentos.

O objetivo deste trabalho é apresentar uma experiência realizada em 2004 com uma turma do curso de Engenharia da Computação do período integral da Faculdade de Engenharia de Sorocaba, onde os alunos desenvolveram um projeto envolvendo as disciplinas de Estrutura de Dados e Algoritmos e Linguagem de Programação II (abordagem em Paradigmas de Programação). O curso trabalha com regime anual e a atividade foi realizada no 2º semestre do período letivo. A experiência teve um período de planejamento da atividade, realizado pelos docentes responsáveis pelas disciplinas envolvidas, antes da atividade ser aplicada. Os docentes tinham como foco principal apresentar aos alunos como os conceitos estudados em disciplinas diferentes, mas que fazem parte de um mesmo eixo de conhecimento, poderiam se integrar de forma prática.

2. ENSINO DA PROGRAMAÇÃO DE COMPUTADORES

Diversas são as estratégias que podem ser adotadas no ensino de programação num curso de graduação na área de computação sendo que cada vez mais, docentes buscam maneiras motivadoras para inserir aos alunos os conceitos que envolvem as técnicas de programação.

Segundo GREENING (2000), a estratégia mais comum de ensino de programação se constitui em aulas teóricas que são complementadas por laboratórios onde os alunos aplicam, através de uma linguagem de programação, os conteúdos abstratos aprendidos.

Embora esta seja uma estratégia importante dentro do processo de ensino da programação ela não deve ser isolada, pois podem acabar por gerar um processo de reprodução daquilo que foi transmitido em sala de aula em detrimento com a real construção do conhecimento.

JENKINS (2002) aborda que as dificuldades do ensino de programação podem estar em fatores cognitivos que envolvem tanto diferentes estilos de aprendizagem bem como a motivação dos alunos diante do objeto a ser aprendido.

Visualizar a construção de programas em diversos paradigmas de linguagens de programação não é transparente ao aluno. As dificuldades encontradas durante o processo de aprendizagem são reflexos de diversos problemas como lógica de programação, a não familiaridade com um determinado paradigma, entre outros. Estas barreiras podem se tornar ainda maiores quando a abordagem está relacionada a conceitos da programação orientada a objetos, JONES (2003), JENKINS (2002) e THRAMBOULIDIS (2003). É fundamental haver, durante o processo de aplicação dos conceitos, um maior foco na solução do problema, e não na sintaxe da linguagem utilizada.

Segundo as Diretrizes Curriculares de Cursos da Área de Computação e Informática, “o estudo de programação não se restringe ao estudo de linguagens de programação. As linguagens de programação constituem-se em uma ferramenta de concretização de software, que representa o resultado da aplicação de uma série de conhecimentos que transformam a especificação da solução de um problema em um programa de computador que efetivamente resolve aquele problema.” Esta abordagem reforça o princípio de que o importante para o aluno é saber aplicar de maneira coerente as técnicas e abordagens estudadas através dos paradigmas de programação.

Segundo AZEREDO (2000), “o conhecimento e o domínio das estruturas de dados e suas representações físicas são importantes para selecionar, de acordo com a aplicação, qual a forma mais eficiente e eficaz de representar os dados”. O ensino das estruturas de dados deve estar apoiado na forma de como os dados são estruturados durante a solução de um problema,

mostrando que algoritmos e estruturas de dados devem se complementar para obtenção de um produto final.

Normalmente, o trajeto a ser percorrido pelo aluno durante o ensino da programação, adotado pelos diversos cursos de graduação, se inicia com a aprendizagem da programação imperativa, pois a mesma possibilita que a lógica de programação possa ser explorada nos alunos com maior facilidade. Quando o aluno começa a ter contato com uma linguagem orientada a objetos ele sente muitas vezes dificuldades em transpor as barreiras que diferenciam os estilos de programação imperativa e orientada a objetos.

Este trajeto, sobretudo, ainda é o mais adotado, pois segundo BURTON E BRUHN (2003), o paradigma orientado a objetos utiliza estruturas existentes no paradigma imperativo. Caso seja invertida a ordem no processo de aprendizagem destes paradigmas, seria necessário explorar os aspectos estruturais imperativos primeiro para depois introduzir os conceitos da orientação a objetos propriamente dito.

Além disto, dar soluções através de uma linguagem procedural é muito mais fácil para o aluno. O aluno sente-se mais à vontade em manipular as estruturas de dados e os algoritmos através de uma linguagem procedural. Embora a linguagem orientada a objetos apresente a forma nativa de modelar as estruturas.

Sintetizando de maneira objetiva as diretrizes que regem o ensino da programação em conjunto com as necessidades de formação profissional torna-se claro que a multidisciplinaridade entre diferentes conteúdos que compõem o eixo do ensino de programação é uma forma de apresentar ao aluno como diversos conteúdos de disciplinas diferentes podem ser integrados para a solução de um problema. Ou seja, dentro do aspecto abordado neste trabalho é possível concluir que o conteúdo abordado na disciplina de linguagem de programação poderá ser o ferramental para se obter a solução de um determinado problema utilizando-se estruturas de dados e algoritmos.

3. INTERDISCIPLINARIDADE E MULTIDISCIPLINARIDADE NO ENSINO SUPERIOR

Segundo OLGUIN (2002), interdisciplinaridade consiste na prática da interação entre os componentes do currículo de um curso. A interdisciplinaridade permite que os envolvidos (alunos e docentes), estabeleçam uma relação entre os conteúdos de diferentes disciplinas.

Um projeto multidisciplinar é aquele que envolve mais de uma disciplina dentro do seu processo de estudo e execução. Uma atividade multidisciplinar possibilita a exteriorização de aspectos que possuem ou não relação entre os conteúdos envolvidos no trabalho, contribuindo para que a interdisciplinaridade ocorra.

A fragmentação de conteúdos abordados em diferentes disciplinas normalmente não permite que o aluno aflore para novas aplicações de conhecimento que extrapolem a visão unificada. Cada vez mais se deseja do aluno que ele tenha uma compreensão da importância da interação e transformação recíprocas entre as diferentes áreas de sua formação.

Quando não há incidências de interdisciplinaridade em um curso de graduação o aluno pode ter a sensação de que os conteúdos que ele aprendeu não são suficientes para sua vida profissional, já que o mercado de trabalho globalizado utiliza os conteúdos de forma integrada. Além disto, o processo de mudanças tecnológicas dadas pelo processo acelerado de disseminação de informação intensifica a necessidade de uma visão prática da interdisciplinaridade de conteúdos, aborda SIQUEIRA (1999).

Deve estar claro o ensino de disciplinas isoladas também são importantes para a formação do aluno. Porém, em certos momentos da formação do graduando é fundamental que ele visualize a conexão e integração dos conceitos. Para SIQUEIRA E PEREIRA (1995), “trabalhar com a interdisciplinaridade não significa negar as especialidade e objetividade de cada ciência. O seu sentido, reside na oposição da concepção de que o conhecimento se

processa em campos fechados em si mesmo, como se as teorias pudessem ser construídas em mundos particulares sem uma posição unificadora...”

4. O PROJETO MULTIDISCIPLINAR

Refletindo sobre a demanda de trabalhos e aplicações que permitissem uma maior integração entre diferentes conteúdos de um curso de graduação em Engenharia da Computação os docentes das disciplinas de Estrutura de Dados e Algoritmos e de Linguagem de Programação II (com abordagem em Paradigmas de Programação), ministradas no mesmo período letivo, decidiram desenvolver um projeto de integração entre as duas disciplinas.

Este trabalho apresenta uma experiência multidisciplinar dentro do eixo curricular da programação de computadores, já que o aluno utilizará as disciplinas de Linguagem de Programação II e Estrutura de Dados e Algoritmos para realizar a atividade. Porém, tal multidisciplinaridade contribuiu para que discussões comparativas sobre as estruturas das linguagens em comparação com a construção de estruturas de dados pudessem desencadear o processo de interdisciplinaridade.

Num primeiro momento foram realizadas algumas reuniões onde os docentes das disciplinas envolvidas discutiram qual seria o objetivo do trabalho, o que se pretendia que os alunos desenvolvessem como habilidades ao final da atividade. Conclui-se que o objetivo principal seria motivar os alunos a buscarem soluções aplicando estruturas de dados para problemas descritos por eles próprios, utilizando-se de uma linguagem orientada a objetos como ferramenta para desenvolvimento desta solução. A partir disto, tornou-se necessária a realização de algumas reuniões de planejamento da atividade.

4.1. Planejamento da atividade

Durante o planejamento procurou-se descrever quais seriam os passos para a execução da atividade com os alunos e como estes seriam conduzidos, pois segundo NOGUEIRA (2001), o docente que participa de um projeto multidisciplinar deve agir como um facilitador conduzindo os alunos no processo de busca das informações necessárias.

Observou-se também que além de apresentar os objetivos da atividade aos alunos, seria fundamental deixar claro desde o início do trabalho quais seriam os itens avaliados quando o projeto fosse entregue. Neste momento os docentes responsáveis pela atividade discutiram sobre a importância de demonstrar ao aluno não somente aplicações práticas do ensino, mas também de estimular neles a busca pela informação através da pesquisa e da contextualização dos resultados pesquisados. Segundo DEMO (2002), é importante desenvolver no aluno a percepção de que a pesquisa não deve ser um objeto de ensino, mas sim um parceiro constante na construção de seu conhecimento. Desta forma, decidiu-se também agregar ao projeto a formalização escrita da solução e dos itens que auxiliaram o aluno na mesma.

Para concluir o trabalho o aluno deveria realizar uma apresentação oral do projeto onde ele defenderia a solução dada ao problema, explanando aos docentes das disciplinas envolvidas todo o percurso de pesquisa e de aplicação prática do seu projeto.

Os docentes formalizaram os aspectos que seriam adotados para a condução da atividade dando origem ao planejamento do projeto, conforme descrito na Tabela 1.

Após definir como o projeto seria conduzido e quais seriam os passos e suas respectivas metas, os docentes se debruçaram no planejamento dos itens que seriam utilizados na avaliação do projeto. Estes itens seriam apresentados aos alunos para que eles tivessem consciência não somente da condução da atividade, mas também sobre quais pontos seriam avaliados. Os aspectos que seriam observados durante a avaliação do projeto e o objetivo de cada um deles são apresentados na Tabela 2.

Tabela 1 - Planejamento do projeto

Objetivo do Projeto: apresentar a solução para um problema através de uma linguagem orientada a objetos utilizando estruturas de dados e algoritmos.	
Disciplinas Envolvidas: Estruturas de Dados e Algoritmos e Linguagem de Programação II.	
Etapas	Meta
1: Situação Problema	Definir qual a situação problema que o aluno desenvolverá o projeto
2: Conceitos Utilizados	Determinar quais estruturas de dados e algoritmos seriam utilizados na solução do problema
3: Linguagem de Programação	Definir qual linguagem orientada a objetos seria utilizada para implementação da solução
4: Desenvolvimento da Solução	Desenvolvimento prático do trabalho
5: Entrega do Projeto	Entrega do trabalho escrito, da codificação e apresentação oral

Tabela 2 - Planejamento da avaliação final do projeto

Tópico Verificado		Objetivo
Desenvolvimento prático	Adequação da estrutura de dados e do algoritmo ao problema	Constatar que o aluno realizou uma escolha adequada para a solução do problema proposto
	Compreensão da estrutura de dados e do algoritmo utilizado	Verificar se o aluno pesquisou e estudou os conceitos que envolvem as estruturas de dados e os algoritmos adotados
	Inovação e sugestões à solução do problema	Analisar se o aluno desenvolveu novas habilidades no campo da programação durante o desenvolvimento da solução do problema
	Utilização de recursos provenientes da linguagem orientada a objetos adotada	Verificar se o aluno tem o domínio necessário da linguagem que escolheu e se está apto a explorar os recursos provenientes de uma linguagem orientada a objetos
	Aplicação dos conceitos de reutilização de código	Constatar que o aluno consegue aplicar os conceitos que envolvem a reutilização de código dentro de um problema
	Uso do encapsulamento (grau de encapsulamento)	Constatar que o aluno teve a preocupação de modularizar a solução e deixá-la encapsulada
Trabalho escrito	Clareza e organização	Verificar se o aluno consegue descrever o processo de desenvolvimento do projeto de forma objetiva e clara
	Descrição de todos os recursos utilizados (classes, algoritmos, etc)	Verificar se o aluno consegue contextualizar os conceitos utilizados de forma prática
Apresentação oral	Clareza e objetividade	Constatar que o aluno consegue transmitir o que abordou em seu projeto
	Abordagem completa do trabalho	Constatar que o aluno se preocupou em apresentar todos os tópicos que foram fundamentais para se obter a solução do problema
	Respostas as perguntas da banca	Verificar se o aluno realmente compreendeu o trabalho desenvolvido e se está apto a defender as decisões tomadas durante o projeto

Procurou-se através dos itens descritos na Tabela 2 percorrer o trabalho do aluno como um todo no momento da avaliação, demonstrando a ele que no desenvolvimento de um projeto todo o processo é importante, desde a contextualização escrita, passando pelo desenvolvimento prático até a apresentação oral. Objetivou-se também apontar que é fundamental ter domínio do assunto para poder defender suas idéias e para dar soluções para os problemas. Domínio este obtido através de pesquisas sobre o assunto.

Ficou acordado que o trabalho seria individual permitindo aos alunos se focarem no tema que mais os interessasse. Após realizar o planejamento da atividade e determinar os itens a serem avaliados, o próximo passo foi apresentar o planejamento aos alunos.

4.2. Apresentação da atividade aos alunos

A apresentação da atividade foi realizada pelos docentes das duas disciplinas onde cada um dos itens descritos no planejamento foi explorado em detalhes. Foi destacada a importância da relação entre os conteúdos abordados nas disciplinas no âmbito de desenvolvimento da aprendizagem durante a explanação do objetivo do projeto destacando o aspecto multidisciplinar da atividade.

Em relação à etapa de definição da situação problema foi determinado aos alunos que eles deveriam escolher qual problema gostariam de resolver. O objetivo era que o próprio aluno se tornasse responsável pela escolha do tema que iria trabalhar, buscando assuntos que o interessasse. Se os docentes determinassem o tema o aluno ficaria numa posição passiva dentro do contexto de sua própria aprendizagem.

Após definir o problema a ser resolvido os alunos deveriam definir quais estruturas de dados e algoritmos seriam necessários para se obter a solução do problema. Neste momento, o aluno teria que estudar de forma mais aprofundada os assuntos relativos à disciplina de Estrutura de Dados e Algoritmos, pesquisando em outras fontes além daquelas utilizadas durante as aulas.

Para implementar a solução o aluno deveria escolher uma linguagem orientada a objetos. Embora Java seja a linguagem utilizada pela disciplina de Linguagem de Programação II, permitiu-se que o aluno optasse pela linguagem que achasse mais adequada. Desta maneira o aluno se tornaria parte ativa das escolhas realizadas durante o projeto e teria a opção, também, aprender uma outra linguagem caso lhe interessasse.

Além disto, seria uma maneira da docente da disciplina de Linguagem de Programação II de verificar se os alunos estariam de fato aprendendo o paradigma orientado a objetos e não uma linguagem em específico.

Definindo as estruturas de dados e algoritmos e escolhendo a linguagem o aluno poderia iniciar a fase de desenvolvimento prático do projeto, implementando a solução para o problema proposto por ele. Além disto, ele deveria montar um documento descritivo do projeto que contivesse os itens relevantes para o seu desenvolvimento: situação problema, estrutura de dados e algoritmos adotados, uma breve descrição da linguagem escolhida, modelo da solução dada e a conclusão do trabalho.

A apresentação oral seria o marco final do projeto, onde o aluno deveria apresentar todas as etapas desenvolvidas justificando suas escolhas e demonstrando o funcionamento do sistema.

Foram apresentados também os tópicos que seriam avaliados dentro do projeto, destacando a importância destes itens na formação do aluno e no desenvolvimento de projetos.

Ao final da explanação os docentes puderam observar que os alunos sentiram-se bastante motivados com o desenvolvimento de uma atividade que provesse através conteúdos diferentes uma visão integradora e não mais fragmentada. Além disto, foi possível observar que eles se sentiram de fato ativos dentro do seu próprio processo de aprendizagem, já que iriam realizar escolhas durante o desenvolvimento do projeto.

4.3. A execução da atividade

O desenvolvimento da atividade iniciou-se logo após a apresentação do planejamento para os alunos. Os alunos tiveram uma semana para definirem qual seria a situação problema que

iria trabalhar. Durante esta etapa os alunos foram orientados pelo docente da disciplina de Estrutura de Dados e Algoritmos, que em alguns momentos fez sugestões de áreas onde eles poderiam encontrar situações problemas que se encaixassem dentro do contexto da atividade.

Após a definição do problema os docentes envolvidos na atividade analisaram as propostas dos alunos sugerindo, quando necessário, algumas modificações. Alguns dos temas escolhidos pelos alunos foram:

- Simulação de roteamento de pacotes em uma rede de computadores através de grafos.
- Calculadora de polinômios.
- Ferramenta de apoio ao ensino da recursividade através da construção de árvores.
- Crescimento da região segmentada de imagens.
- Simulação da estrutura de índices de banco de dados utilizando árvores binárias.

Assim que o tema foi aprovado os alunos iniciaram o processo de pesquisa dos recursos que precisariam para dar a solução para o problema. Nesta etapa os dois docentes trabalharam na orientação de como conduzir o processo de pesquisa, que culminou na coleta de diversas fontes de informação contendo dados relevantes sobre algoritmos e estrutura de dados necessários para cada projeto.

O passo seguinte se constitui na definição da linguagem de programação orientada a objetos que seria utilizada para o desenvolvimento da solução. Nesta etapa a docente da disciplina de Linguagem de Programação II orientou os alunos para as possíveis opções em relação ao desenvolvimento do tema. O fim desta etapa se deu com a definição das linguagens. Além da linguagem utilizada na disciplina, Java, outras também utilizadas pelos alunos durante a fase de implementação foram: C++ e C#.

Munidos dos recursos estruturais e da linguagem que serviria como ferramenta para solução os alunos puderam iniciar o procedimento de implementação do projeto. Durante esta fase os alunos trabalharam sozinhos na codificação da solução. Porém, a docente de Linguagem de Programação II procurou conduzir a estruturação do trabalho escrito. Através de orientações realizadas durante as aulas os alunos tiveram uma visão de como associar a pesquisa realizada com a solução dada através da implementação prática. Definiu-se um modelo para o desenvolvimento do trabalho escrito contendo:

- **Introdução:** deveria situar o leitor sobre a área em que o problema estava inserido, descrevendo o objetivo do projeto e qual a motivação para o desenvolvimento do mesmo.
- **Corpo:** conteria o desenvolvimento do tema, descrevendo as estruturas de dados e os algoritmos adotados, bem como a linguagem utilizada. Neste ponto o aluno deveria apresentar também um modelo da solução proposta. Não sendo necessário agregar ao texto a codificação da solução, já que a mesma seria entregue através de uma mídia qualquer.
- **Conclusão:** apresentar o resultado obtido e quais foram as dificuldades encontradas no desenvolvimento do projeto.

Além disto, a docente de Linguagem de Programação II orientou os alunos sobre a apresentação oral do trabalho que deveria ter a duração de 20 minutos. Neste momento, os alunos foram motivados a planejar a sua própria apresentação, trabalhando com os aspectos relevantes de forma a demonstrar todo o trajeto percorrido durante o desenvolvimento do projeto. No final do período letivo os alunos realizaram a apresentação dos projetos para os docentes envolvidos.

4.4. A avaliação da atividade

O processo de avaliação de uma atividade deve sempre se constituir em um procedimento de mão-dupla, onde alunos e docentes extraem os pontos positivos e negativos.

Durante a apresentação oral os alunos já puderam perceber aquilo que os trabalhos tinham como aspecto a ser destacado ou criticado, sempre de forma construtiva. Os docentes procuraram sempre conduzir as perguntas ou as críticas de maneira que o próprio aluno concluísse aquilo que estava sendo apontado como uma falha. Também procuraram exaltar os aspectos positivos de cada trabalho, fazendo com que os alunos valorizassem ainda mais a atividade desenvolvida.

Foi também durante o momento da apresentação que os docentes abordaram o aspecto interdisciplinar da atividade apontando as relações e não relações entre as disciplinas que compunham o projeto.

Após a apresentação oral os docentes se reuniram para analisar o trabalho escrito e a codificação dos alunos trocando impressões sobre os temas e resultados. Esta atividade foi fundamental para a determinação da avaliação final do aluno, já que os docentes puderam compartilhar os pontos que observaram durante o desenvolvimento da atividade toda refletindo sobre os resultados apresentados pelos alunos.

5. CONCLUSÕES

Observando os trabalhos dos alunos durante todo o período em que atividade foi desenvolvida, foi possível constatar que houve um grande esforço por parte dos estudantes para que o objetivo proposto fosse atingido comprovando o aspecto motivador que uma atividade multidisciplinar pode desencadear. Outro ponto bastante interessante foi o fato de alguns alunos terem escolhido linguagens que não conheciam para realizar a implementação do projeto, demonstrando grande flexibilidade dentro do contexto de crescimento profissional.

Foi possível observar, através da análise dos relatórios e das respostas às perguntas durante a apresentação dos trabalhos, o amadurecimento dos alunos em relação à aplicação dos conceitos de estruturas de dados e de orientação a objetos.

Dificuldades práticas da orientação a objetos como a modelagem do número adequado de classes, utilização eficiente dos mecanismos de herança, e compreensão clara de conceitos como agregação e composição, foram expostas pelo experimento realizado, e esclarecidas pelos avaliadores, consolidando e lapidando esses conceitos nos alunos.

A compreensão e abstração necessária relativos aos conceitos de estruturas de dados e algoritmos desenvolveram-se mais facilmente através da experiência prática dos alunos ao implementar problemas clássicos desta disciplina e encontrar as dificuldades intrínsecas aos mesmos.

Do ponto de vista da interação dos docentes com a atividade, conclui-se que há a necessidade de lapidação de alguns pontos, como, por exemplo, dar um tempo maior para que os alunos possam definir a situação problema. Desta forma, o aluno poderá realizar uma pesquisa sobre o tema que deseja se aprofundar elaborando um projeto mais contextualizado.

Outro ponto bastante importante foi que alguns alunos conseguiram vislumbrar dentro dos conteúdos estudados em seus projetos outras vertentes de pesquisa, demonstrando que integração de conteúdos pode permitir inclusive uma nova busca por outros conhecimentos.

Tanto o caráter multidisciplinar como o interdisciplinar da experiência permitiu que o aluno não visualizasse a aprendizagem dos conceitos das duas disciplinas envolvidas de forma fragmentada. Permitiu o afloramento de processos internos de amadurecimento e de desenvolvimento dentro do contexto da aprendizagem da programação, que só podem ocorrer quando o aluno interage com aspectos múltiplos dentro de um eixo de aprendizagem.

REFERÊNCIAS BIBLIOGRÁFICAS

AZEREDO, P A. Uma proposta de Plano Pedagógico para a matéria de Programação. In: II CURSO: QUALIDADE DE CURSOS DE GRADUAÇÃO DA ÁREA DE COMPUTAÇÃO E INFORMÁTICA, WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2000, Curitiba. **Anais** Editora Champagnat, 2000.

BURTON, P J; BRUHN, R. E. Teaching Programming in the OOP Era. **ACM SIGCSE Bulletin**. v. 35, i. 3, 2003.

DEMO, P. **Educar pela pesquisa**. Campinas: Editora Autores Associados, 2002.

Diretrizes Curriculares de Cursos da Área de Computação e Informática. MEC - Secretaria De Educação Superior.

<http://portal.mec.gov.br/sesu/index.php?option=content&task=view&id=430&Itemid=420>

MASETTO, M T. Aula na Universidade. **Didática e Interdisciplinaridade**. São Paulo: Editora Papirus, p. 179-192, 1998.

GREENING, T. Emerging Constructivist Forces in Computer Science Education: Shaping a New Future?. **Computer Science Education in the 21st Century**. New York: Springer-Verlag, p. 47- 80, 2000.

JENKINS, T. On The Difficulty of Learning to Program. **3rd Annual LTSNICS Conference**. United Kington, 2002.

JONES, R ET AL. ObjectWorld: Helping Novice Programmers to Succeed Through a Grafical Objects-first Approach. **4th Annual LTSN-ICS Conference**. United Kington, 2003.

NOGUEIRA, N R. **Interdisciplinaridade Aplicada**. São Paulo: Ed. Érica, 2001.

OLGUIN, G. **Interdisciplinaridade**. Disponível em: www.campogeral.com.br, 2002. Acessado em: 20/04/2004.

SEBESTA, R W. **Conceitos de Linguagens de Programação**. Porto Alegre: Bookman, 2002.

SIMONSON, M ET AL. **Teaching and Learning at a Distance. Foundations of Distance Education**. United States of America: Prentice Hall, 2000.

SIQUEIRA, H S G. **Prática Interdisciplinar na Universidade: descontextualização**. Disponível em: <http://geocities.yahoo.com.br/holgonsi/interdisciplinar2.html>, 1999. Acessado em 15/08/2004.

SIQUEIRA, H S G; PEREIRA, M A. **Interdisciplinaridade como superação da fragmentação**. Disponível em: <http://geocities.yahoo.com.br/holgonsi/interdisciplinar3.html>, 1995. Acessado em 15/08/2004.

THRAMBOULIDIS, K ET AL. Teaching Advanced Programming Concepts in Introductory Computing Courses: A Constructivism Based Approach. **Internacional Conference on Engineering Education**. Spain, 2003.

MULTIDISCIPLINAR PROJECT: A PRATICAL EXPERIENCE ON PROGRAMMING TEACHING IN A COMPUTER ENGINEERING COURSE

***Abstract.** This article presents a teaching experience involving two subjects in the computer programming area. A group of Computer Engineering students has been observed during the learning process. The students were enrolled in the Data Structures and Algorithms and Programming Language II courses. The main goal was to motivate them to develop, through an oriented object language, solutions of problems proposed by themselves, applying in the solution the methods studied in the Data Structures and Algorithms subjects.*

***Key-words:** Multidisciplinar project, Programming teaching, Interdiscipline.*