

UMA ABORDAGEM PARA O ENSINO DE PROGRAMAÇÃO NAS ENGENHARIAS

Dalton Vinicius Kozak - dvkozak@ccet.pucpr.br

Henri Frederico Eberspächer - henri@ppgia.pucpr.br

Pontifícia Universidade Católica do Paraná, Laboratório de Mídias Interativas
Rua Imaculada Conceição, 1155 - 80215-901 - Curitiba - PR

***Resumo.** O programa das disciplinas de algoritmos e programação para os diversos cursos de engenharia tradicionalmente usam um modelo comum, oriundo dos cursos da área de informática, como engenharia/ciência da computação ou análise de sistemas. Embora a base teórica possa ser a mesma, o enfoque ou objetivo é diferente entre esses cursos. Nesse trabalho será discutida a forma como está sendo conduzido o ensino de algoritmos e programação em alguns cursos de engenharia na Pontifícia Universidade Católica do Paraná. A definição do conteúdo, a forma de abordagem do ensino, as ferramentas utilizadas no ensino de programação, bem como a utilização de um ambiente colaborativo de ensino através da Internet, desenvolvido na PUC em parceria com a Siemens do Brasil, serão focos de discussão.*

***Palavras-chave:** Ensino de algoritmos, Ensino de programação, Processo pedagógico.*

1. INTRODUÇÃO

Não se pode discutir que a padronização é um passo importante no processo de melhoria de qualidade, estabelecendo referenciais para controle de processos/produtos e propiciando diminuição de custos, muitas vezes um mesmo recurso podendo ser reutilizado em diferentes contextos. Porém, isso não deve ser feito perdendo de vista o objetivo, ou o cliente final, do processo ou produto, pois embora o fabricante de sapatos saiba que pode produzir com menor custo e melhor qualidade um só modelo de determinado calçado apenas no tamanho 40, certamente nem todos os clientes calçam esse número ou gostam de tal modelo.

Guardadas as devidas diferenças, o mesmo raciocínio pode se aplicar na elaboração do conteúdo de disciplinas nos cursos de graduação, especialmente aquelas que são comuns a diversos cursos, como é o caso do curso de algoritmos e programação: nem todos os cursos “calçam” uma disciplina de algoritmos “tamanho 40”. Entretanto, não é difícil encontrar nas diversas instituições de ensino alunos de diferentes cursos de ciências exatas, da área de informática ou não, cursando uma disciplina de algoritmos e programação com mesmo conteúdo e estruturação. Antes da implantação do novo projeto pedagógico, essa também era a prática vigente em nossa instituição (PUC PR). Embora exista ganhos em se padronizando a disciplina, como a intercambialidade da disciplina entre os diversos cursos e a manutenção do

mesmo material didático, perde-se no objetivo final, que é proporcionar ao aluno a informação na quantidade, qualidade e profundidade adequada ao seu curso.

Nesse trabalho são abordados alguns aspectos considerados como relevantes na montagem de uma disciplina de algoritmos e programação adequada às necessidades de um curso de engenharia. Tais aspectos abordam os seguintes pontos:

- O perfil dos estudantes da área de informática e das outras áreas de exatas.
- O foco da disciplina: programação é meio ou é fim?
- Escolha das ferramentas de programação.
- A metodologia e teorias educacionais envolvidas no processo de ensino/aprendizagem.

2. ASPECTOS RELEVANTES

2.1 Perfil dos Estudantes

É possível notar que, de uma forma geral, os estudantes que procuram cursos na área de informática geralmente são aqueles que já possuem alguma familiaridade com a utilização de computadores, motivo pelo qual muitos optam por tais cursos. Não é raro encontrar entre estes estudantes aqueles que adentram a universidade já com algum conhecimento de programação. Por outro lado, apesar do grande número de escolas de 2º grau e lares que já possuem microcomputadores, é surpreendente ainda observar uma grande quantidade de alunos dos cursos de engenharia (civil, elétrica, mecânica, etc..) que chegam à universidade com conhecimento precário na utilização desses equipamentos. Os autores estimam que não chega a 5-10% a quantidade de alunos que começam o curso com um conhecimento de informática que pode ser considerado razoável.

Portanto, em termos médios, há uma diferença entre a bagagem de conhecimento trazida pelos estudantes que vem especificamente para a área de informática e aquela trazida pelos estudantes de cursos voltados para as outras áreas das ciências exatas. Este único aspecto já é suficiente para estabelecer uma diferenciação na elaboração do conteúdo de uma disciplina de algoritmos e programação, exigindo uma ênfase maior em tópicos iniciais de informática, e um tratamento mais demorado na parte de algoritmos e a lógica associada, menos desenvolvida nesse segundo grupo de estudantes.

2.2 Foco da Disciplina

Programação é meio ou é fim? Para estudantes dos cursos na área de informática, um bom conhecimento e aprofundamento nas questões de programação é importante, já que muitos, em seu exercício profissional, provavelmente atuarão como especialistas em programação procurando atender diferentes demandas provenientes de várias áreas de conhecimento. Nesse caso, técnicas mais avançadas de programação são necessárias para melhor poder resolver os diversos tipos de problemas, que podem envolver tanto aplicações ditas científicas, como aplicações gráficas e manipulação de banco de dados. Nesse contexto, programação é fim.

Para os outros cursos de engenharia, o escopo de conhecimento necessário não precisa ser tão amplo e pode focar nos problemas importantes a serem resolvidos relacionados a cada especialidade em particular. Assuntos como ponteiros, alocação dinâmica de memória, listas encadeadas ou algoritmos de ordenação são dispensáveis (embora interessantes), uma vez que a classe de problemas tratados nas engenharias não dependem essencialmente desse conhecimento. Em boa parte, os algoritmos envolvem problemas de cálculo numérico, como a resolução de sistemas lineares, onde é suficiente o conhecimento das estruturas de controle básicas (seqüência, alternativa, repetição) e a noção de estruturas de dados homogêneas (vetores e matrizes). Nesse contexto, programação é uma ferramenta, um meio, para os estudantes/profissionais resolverem problemas de engenharia específicos.

2.3 As Ferramentas de Programação

Não é raro a escolha da ferramenta de programação a ser utilizado no curso de algoritmos sem maior critério. Consideramos que nessa questão dois pontos principais devem ser analisados: o nível de conhecimento dos alunos e o objetivo do curso.

pode ser classificado como o de iniciantes, e nesse caso há um sentimento de que a escolha da linguagem de programação deve levar em conta esse aspecto. Pascal ou escolhas naturais como uma primeira linguagem a ser ensinada, já que foram criadas com esse objetivo. Mas talvez esse não seja o "x" da questão, uma vez que os autores tem percebido

construção/entendimento do algoritmo propriamente dito, e que a sua tradução para uma linguagem qualquer, incluindo o C, é superável mesmo para o iniciante quando o algoritmo

para o iniciante tenha seus benefícios, o raciocínio algorítmico é o fator preponderante no aprendizado da programação.

através de programação em cada curso de engenharia em particular. Existe uma base de conhecimento que será comum, e que passa pelos algoritmos provenientes do cálculo

utilização de métodos numéricos. Mas existem problemas típicos de cada especialidade. O Eng- Civil faz grande uso de planilhas de cálculo, como no levantamento de custos de obras ou no cálculo de quantidade de material necessário para determinada obra ou construção. O

Elétrico voltado para telecomunicações deve conhecer sistemas operacionais, sistemas

planilhas eletrônicas como o MS-Excel e sua linguagem de programação embutida, o VBA (Visual Basic for Applications) é de grande utilidade, ao passo que para o segundo a

bem como orientação a objeto (C++), é adequada para resolver seus problemas. Para ambas as especialidades, a utilização de ferramentas como SciLab também pode ser bastante útil. Tal análise é aplicável para cada ramo da engenharia.

A Metodologia de Ensino/Aprendizagem e Teorias Educacionais

or trás da prática pedagógica, é necessário um pequeno resumo de algumas das principais abordagens educacionais existentes. Isso pode ser Tabela 1. Outro conceito que se mostra relevante nesse contexto está associado ao tipo de relacionamento que o professor deve manter com seus alunos. Conforme observado (Kozak e Eberspächer, 2000), os conceitos oriundos da análise transacional e da situacional também são relevantes nesse contexto. De forma sucinta, a partir dessas teorias é possível estabelecer a forma como o professor deve conduzir seu relacionamento

a capacitação e disposição, conforme mostrado na Tabela 2. Cruzando as informações

Figura 1, que permite escolher o princípio educacional mais adequado.

Voltando ao item (Perfil dos Estudantes) grupos de estudantes analisados apresentam uma diferença na dimensão de capacitação. Se esta diferença for acompanhada, no mesmo sentido, pela dimensão da disposição, o patamar

de inícios de trabalhos será diferente para os dois grupos: os estudantes da área de informática podem começar um pouco mais à frente na disciplina de algoritmos e programação ¹.

Abordagem Educacional	Tipo de Instrução	Instrução e Manipulação	Realidade a ser construída
Comportamentalismo	A instrução está focada do comportamento do	O instrutor manipula as mudanças reforços seletivos.	são convergentes. Ou seja, realidade sobre a qual a
Cognitivismo	na transmissão do pensamento ao	O instrutor manipula o processo de fornecendo a ele um modelo	As realidades dos aprendizes
Construtivismo	construção mental da aprendiz.	facilitador do conceitual prover atividades de aprendizagem possa construir seu conhecimento.	são divergentes: eles são sua própria realidade.
A técnica ABP²	O aprendizado baseia-se na solução de um problema real por pequenos grupos de discussão conduzido por um “tutor” no papel de “aprendiz sênior”. Esse arranjo promove o processamento da informação pelo grupo.		

Tabela 1. Breve apanhado das principais abordagens educacionais.

ALUNO				PROFESSOR			
MATURIDADE			Nível de Maturidade	Estilo de Liderança	Característica	Relacionamento	
CAPACITAÇÃO Aluno tem os conhecimentos e habilidades necessárias	DISPOSIÇÃO Aluno tem a confiança e o comprometimento necessários						
pouco	Maturidade para o trabalho	raramente	Maturidade Psicológica	M ₁	E ₁	Direção	Líder dá instruções específicas e monitora a performance do subordinado de perto.
algum		algumas vezes		M ₂	E ₂	Coordenação	Líder explica suas decisões e proporciona espaço para discussões.
razoável		frequentemente		M ₃	E ₃	Colaboração	Líder troca idéias com os subordinados e auxilia na tomada de decisões.
Muito		sempre		M ₄	E ₄	Delegação	Responsabilidades pela execução das tarefas e pela tomada de decisões são delegadas aos subordinados

Tabela 2. Relacionamento entre o nível de maturidade dos subordinados/alunos e o estilo de liderança recomendado para o líder/professor.

¹ Essa observação é condizente com que normalmente se verifica na prática.

² Aprendizagem baseada em problemas (PBL - Problem-based Learning).

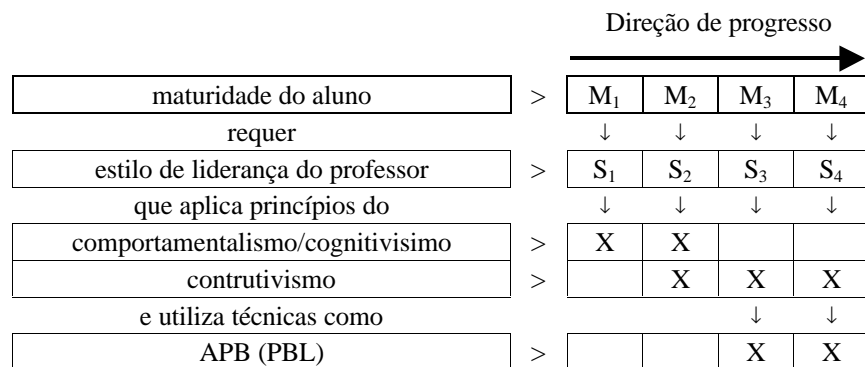


Figura 1. Um pequeno esquema para selecionar uma abordagem educacional.

3. NA PRÁTICA

3.1 Como Chegam os Alunos?

A disciplina de algoritmos e programação geralmente é ministrada no primeiro ou segundo ano dos cursos de engenharia. Fazendo referência à Tabela 2, notamos que o nível de maturidade M₁ é o que mais frequentemente se observa entre os alunos, principalmente quando se trata de alunos do primeiro ano, recém-vindos do nível médio e/ou cursinho, muitos ainda sob a influência da "ditadura do macete", estritamente baseado no procedimento estímulo/resposta, típico do comportamentalismo (Tabela 1). É nesse patamar que boa parte desses alunos estão habituados a trabalhar, com poucas exceções, e é o que se tem observado nas turmas que chegam às nossas salas de aula.

3.2 Aplicando as Abordagens de Ensino/Aprendizagem

O cuidado em situar o nível de maturidade dos alunos é importante no sentido de se adequar o processo pedagógico, pois se for tentado aplicar técnicas de ensino/aprendizagem como o APB logo no início do curso, muito provavelmente os resultados não serão os esperados. Antes de se partir para essas técnicas é necessário amadurecer o aluno, trazendo-o para níveis de conhecimento, habilidade e comprometimento onde a técnica utilizada é bem sucedida. O uso das abordagens educacionais deve ser compatível com a evolução do amadurecimento do aluno. A bem da verdade, cada uma tem o seu tempo, e cabe ao professor, no desenrolar da disciplina, tentar caminhar no sentido do aumento do amadurecimento (direção de progresso; Figura 1) do aluno e conseguir, ao final, empregar técnicas como o APB, mais próximas da forma como se deseja que o egresso trabalhe ao final do curso.

Assim sendo, durante as aulas teóricas os alunos trabalham com material escrito elaborado pelo professor e obtido geralmente através da Internet. A intenção é evitar a perda de tempo tanto do professor quanto dos alunos com a simples cópia de conteúdo da disciplina do quadro para os cadernos. Isso libera tempo para discussões mais detalhadas sobre tópicos importantes e para os exercícios. A dinâmica de aula é baseada em breves explanações teóricas seguidas de exemplos e exercícios propostos. Para estimular a participação e o trabalho em sala, alguns alunos são sorteados para mostrar como fizeram esses exercícios; se estiverem resolvidos corretamente, ou com a solução bem encaminhada, os alunos recebem créditos que posteriormente são adicionados à nota parcial do período. Do ponto de vista das abordagens educacionais, esse procedimento pode ser enquadrado dentro do comportamentalismo e cognitivismo. Exercícios especiais e desafios promovem o aspecto construtivista do processo.

As aulas práticas, em laboratório, são baseadas em um roteiro elaborado pelo professor e estruturado da seguinte forma: objetivo, base teórica, seqüência de atividades e exercício.

Esse roteiro é elaborado de tal forma a acompanhar o conteúdo ministrada em aulas teóricas, porém todo o conhecimento necessário para a prática está contido nesse material. O aluno deve ler esse roteiro antes da prática para poder ter condições de efetuarla com aproveitamento durante o tempo disponível, que é de duas aulas (1h40min). Certamente, nesse procedimento um obstáculo é encontrado: os alunos dificilmente se preparam para as aulas. Se isso não acontece, seu rendimento em aula é comprometido, bem como a sua avaliação. Voltando à Tabela 2, trata-se do um nível de disposição correspondente ao menor nível de maturidade (M_1). Para que a "coisa" funcione, a maturidade do aluno deve ser desenvolvida, e isso acaba por acontecer quando ele percebe que, se não houver seu comprometimento com essa atividade, isso será refletido em sua avaliação.

Por fim, o último quarto do curso é reservado para um trabalho final, em grupo, onde a probabilidade da aplicação da técnica APB tem maior chance de sucesso. Nesse trabalho é proposto aos grupos que eles selecionem um problema a ser resolvido, dentro de certos parâmetros de complexidade e tamanho definidos pelo professor. A partir disso, deve ser construído o algoritmo da solução e escolhida a ferramenta a ser utilizada para implementação desse algoritmo. Uma vez pronto o aplicativo, ele deve ser testado e o resultado apresentado ao professor, que submete cada grupo a uma prova de autoria. Tal procedimento tem apresentado resultados bastante satisfatórios. É bastante animador ver alunos de engenharia civil, que nunca haviam programado antes do curso, apresentar um programa em C onde sistemas lineares de ordem n são resolvidos por métodos iterativos (Gauss-Seidel), incluindo a aplicação de testes de convergência (critério de Sassenfeld) não muito simples, e defendidos de forma convincente na prova de autoria.

3.3 Os Conteúdos e as Ferramentas - Eng^a Civil

O conteúdo básico de uma disciplina de algoritmos e programação permanece praticamente inalterada entre os diversos cursos: conceitos básicos de tipos e variáveis, entrada e saída de dados, estruturas de controle e estrutura de dados, apenas este último com nível de profundidade que pode variar conforme o curso. Para os cursos de engenharia, de uma forma geral, não é necessário ir além de estruturas de dados homogêneas (vetores e matrizes). Esse é o caso do curso de Eng^a Civil em nossa Universidade. Uma carga horária em torno de 120 horas, distribuída em dois semestres, parece ser adequada para o desenvolvimento dessa disciplina.

Para o curso de Eng^a Civil em particular, tem-se utilizado duas ferramentas para programação: o MS-Excel/VBA (Visual Basic for Applications) e um compilador C. O uso da planilha eletrônica, além de vir de encontro com as necessidades específicas desse curso, disponibiliza ao aluno uma ferramenta de programação pouco utilizada, que é o VBA. Através do uso dessa ferramenta é possível implementar algoritmos que empregam as principais estruturas de controle, e com uma vantagem: todos que possuem um computador tem, com grande probabilidade, essa ferramenta disponível em seu equipamento através da planilha Excel. Para tratar problemas envolvendo estruturas de dados homogêneas (solução de sistemas lineares, por exemplo), o compilador C é utilizado. Para completar o conhecimento das ferramentas disponíveis para resolução de problemas através do computador, pretende-se incluir também uma introdução ao MatLab ou SciLab, que são ambientes de programação voltados para resolução de problemas numéricos. No trabalho final, que deve concluir a disciplina, o aluno tem a opção de escolha de uma dessas ferramentas, conforme o problema a ser resolvido.

3.4 Suporte Virtual

Como ferramenta de suporte ao curso tem-se utilizado um ambiente baseado na Web para aprendizagem cooperativa denominado Eureka (<http://www.lami.pucpr.br/eureka>), que é

resultado um projeto de pesquisa desenvolvido no Laboratório de Mídias Interativas (LAMI) da Pontifícia Universidade Católica do Paraná (PUCPR), em parceria com a Siemens do Brasil (Eberspächer *et al.*, 1999). Tal ambiente foi desenvolvido com a intenção de viabilizar a educação e treinamento à distância usando a Internet como meio de criação de comunidades virtuais que participam de cursos que tradicionalmente são presenciais.

Entretanto, apesar da disciplina de algoritmos e programação ser presencial, algumas funcionalidades do Eureka são bastante úteis mesmo nesse contexto, como o edital de avisos, o fórum de discussões e o correio eletrônico, que permitem uma interatividade maior entre os participantes do curso. Esse suporte tem se mostrado útil na troca de informações no que se refere a resolução de exercícios e esclarecimento de dúvidas entre os próprios alunos.

4. COMENTÁRIOS FINAIS

O que se procurou mostrar nesse trabalho foi uma forma pragmática de encarar a preparação de uma disciplina de algoritmos e programação para um curso de engenharia fora da área de informática. Deve-se notar que os pontos considerados nesse processo abrangem análises até certo ponto não convencionais em se tratando desse assunto, porém tem-se mostrado úteis na forma de preparar a abordagem pedagógica da disciplina, não só em termos de conteúdo, mas também levando em conta o perfil dos alunos.

5. REFERÊNCIAS

- Kozak, Dalton V. e Eberspächer, Henri F. *A Non-traditional Approach to Algorithms and Computer Programming Courses in Engineering Education*. In: ICECE'2000 - The International Conference on Engineering and Computer Education, Anais..., São Paulo, 2000
- Eberspächer, Henri F.; Vasconcelos, Cristiano D.; Jamur, José H.; Eleuterio, Marco A. *Eureka: um ambiente de aprendizagem cooperativa baseado na Web para Educação à Distância*. In: Simpósio Brasileiro de Informática na Educação (SBIE), 10., Anais..., Curitiba, 1999.
- O. G. Terra Jr.. *Teorias Pedagógicas*, 1997, URL: <http://www.inf.ufes.br/~tavares/trab2.html>