



**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/UFPG-UFPE

## COMPUTAÇÃO NA ENGENHARIA: ENSINAR A PROGRAMAR E/OU USAR SOFTWARE COMERCIAL?

**Zacarias M. Chamberlain Pravia** – [zacarias@upf.br](mailto:zacarias@upf.br)

Universidade de Passo Fundo, Faculdade de Engenharia e Arquitetura

C.P. 611, Campus I

990001-970, Passo Fundo, RS

**Gilnei A. Drehmer** – [gilnei@upf.br](mailto:gilnei@upf.br)

**Jocarly Patrocínio de Souza** – [jocarly@upf.br](mailto:jocarly@upf.br)

***Resumo:** O ensino de computação nos cursos de engenharia está ficando obsoleto e já não fornece subsídios para a formação dos futuros profissionais. A crescente expansão de sistemas computacionais de baixo custo, seja nas instituições de ensino, seja nos escritórios de estudos e projetos, aumenta a pressão sobre docentes e alunos no sentido de ser incrementada a componente de utilização das novas tecnologias no ambiente universitário. Se, num passado recente, poucos engenheiros trabalhavam regularmente com um computador, nos dias de hoje praticamente todos os recém-formados são colocados em postos de trabalho em que as mais modernas ferramentas da computação são sistematicamente utilizadas. Avaliando os currículos de várias instituições e seus programas para ensino de computação assim como os desafios sobre o que ensinar, como programação de computadores e/ou uso de programas comerciais prontos, este trabalho apresenta o estado atual do ensino de computação, e algumas alternativas a serem consideradas para oferecer um ensino da computação de alto nível, necessário aos engenheiros que irão se formar no século XX e atuar num mercado onde a ciência da informação é tudo.*

Palavras-chave: Computação, Engenharia, Ensino.

## 1. INTRODUÇÃO

A contínua expansão de sistemas computacionais de baixo custo e também de outros de custo elevado, seja nas instituições de ensino, seja nos escritórios de estudos e projetos, aumenta a pressão sobre docentes e alunos no sentido de se incrementar a componente de utilização das novas tecnologias no ambiente universitário. Se, num passado recente, poucos engenheiros trabalhavam regularmente com um computador, nos dias de hoje praticamente todos os recém-formados são colocados em postos de trabalho onde as mais modernas ferramentas da computação são sistematicamente utilizadas. Assim, levanta-se a seguinte questão: o ensino da utilização dessas novas tecnologias deve dar-se no curso universitário ou nas organizações empregadoras? Caso se opte pela primeira hipótese, será necessário sacrificar o ensino das matérias tradicionais para abrir espaço para o ensino das novas tecnologias.

A vantagem mais evidente dessa opção é a de que o aluno se estará bem preparado para ser produtivo já no seu primeiro emprego. Em contrapartida, a desvantagem é a necessidade de sacrificar o ensino dos assuntos mais teóricos e generalistas, limitando-se, assim, as hipóteses de emprego do recém-formado à aplicação das ferramentas que aprendeu a utilizar sem ter conhecido satisfatoriamente os fenômenos em que se baseiam. Por outro lado, caso se ofereça um ensino mais teórico e generalista, está-se a transferir para o empregador os encargos inerentes à formação relacionada com a utilização das tecnologias computacionais. Essa solução apresenta a vantagem de o recém-formado ser dotado de uma cultura mais genérica e poder concorrer a qualquer tipo de emprego, devendo, em seguida, aprender a dominar as ferramentas utilizadas no seu local de trabalho. Desse modo, não existiria na graduação desperdício de tempo para a aprendizagem da utilização de aplicações informáticas com que, provavelmente, o futuro engenheiro jamais irá contatar.

Dessas primeiras considerações gerais afigura-se como mais razoável defender uma situação de equilíbrio entre os dois extremos, que consiste em não fornecer um ensino limitado apenas a noções teóricas desprovidas de aplicação prática, nem limitar os cursos ao treino de utilizadores de equipamentos que apenas necessitam memorizar seqüências de procedimentos automatizados. Deve, portanto, prevalecer um razoável equilíbrio entre as duas situações, com um maior inclinação para o ensino dos fundamentos genéricos.

Grande número de autores observa a necessidade de serem alteradas as características do ensino de computação. OLIVEIRA & ALVES (2001) apresentam como alternativa o uso de técnicas modernas de programação orientada a objetos, as quais, contudo, são de difícil implementação em virtude do acentuado desinteresse com a relação ao ensino de computação, como FERREIRA (2000) relata sobre os alunos da Engenharia da UFOP. Por outro lado, FILHO (2001) afirma que há necessidade de ensinar a lógica da programação, pois o conhecimento não está pronto em nenhuma área da engenharia e, embora exista grande quantidade de programas comerciais, haverá sempre casos em que se exigirá do desenvolvimento de um novo programa com uma nova lógica, talvez provinda de novas pesquisas. Por isso, o estudante de engenharia deve estar preparado para essa possibilidade.

Existe, ainda, a possibilidade de usar programas educacionais no contexto teórico apresentado por PRAVIA & KRIPKA (1999), porém o número de softwares educacionais para engenharia é reduzido, em razão do pouco atrativo para desenvolver esse tipo de ferramentas.

Uma das alternativas bem-sucedidas é apresentada por BARROS & ZAMBONI & PAMBOUKIAN (2004), pela qual o ensino de computação foge do padrão histórico do ensino de Fortran em ambientes de texto e passa a usar linguagens com interfaces mais chamativas (DELPHI, C++), até mesmo desenvolvendo livros de ensino de programação

específicos para ensino de computação de engenharia, textos inexistentes nessa área, à exceção dos velhos algoritmos numéricos desenvolvidos em Pascal ou Fortran.

Uma definição bastante acertada do que deve ser o ensino de computação nas engenharias é dada por BARROS & ZAMBONI & PAMBOUKIAN (2004): formação do raciocínio lógico-matemático num ambiente de programação agradável.

Observa-se que apenas citamos os autores referindo-se ao ensino formal de programação de computadores. A maioria dos alunos na graduação se questiona sobre o porquê de aprender a programar, se a maior parte do conhecimento que vai ser utilizado na sua vida profissional já está pronta e com atualização de versões sendo apresentadas constantemente. Nesse sentido, este trabalho tem por objetivo discutir o estado atual do ensino de computação e a forma de equilibrar ou apresentar alternativas entre o uso de programação própria e o de programas prontos.

## **2. SITUAÇÃO DO ENSINO DE COMPUTAÇÃO**

Aproveitando a comodidade de acesso a dados oferecidos pela internet, foram pesquisados 37 cursos de engenharia (elétrica, civil e mecânica) localizados na região Sul e Sudeste do Brasil. Um panorama resumido desse levantamento mostra que a maior parte desses cursos tem nos seus currículos uma ou duas disciplinas de computação, sob as seguintes denominações:

- Ferramentas Computacionais;
- Programação em Engenharia I e II;
- Introdução à Computação;
- Informática Básica;
- Programação de Computadores I e II;
- Modelagem Computacional na Engenharia;
- Introdução à Ciência da Computação
- Introdução à Computação na Engenharia.

Essas disciplinas estão situadas nos três primeiros semestres do curso e, em geral, a maioria possui uma única disciplina de ensino de computação. A maior ocorrência desse tipo de disciplina única é no segundo ou terceiro semestre do curso.

Quanto ao conteúdo oferecido, um resumo dos programas de disciplinas consultados engloba os seguintes assuntos:

- uso de computadores (hardware);
- arquitetura de computadores;
- uso de editores de texto;
- uso de planilha de cálculo;
- lógica de programação (algoritmos);
- ensino formal de uma linguagem de programação.

As linguagens para ensino de programação usadas são Pascal, Fortran e C na maior parte dos cursos consultados com programação sequencial; poucos são aqueles que utilizam o enfoque de programação orientada a objetos e linguagens com ambientes visuais (Delphi, C++, Visualbasic). Observa-se que quase todos os cursos de engenharia elétrica apresentam como linguagem de ensino o C, já que é uma necessidade natural considerando as metas e necessidades do curso.

Verifica-se que as disciplinas acontecem nos três primeiros semestres, quando a educação profissional específica ainda não ocorreu. Por isso, fica limitado o número de aplicações práticas a serem apresentadas nas disciplinas de ensino de computação.

Questionando alunos de vários desses cursos sobre o uso das técnicas aprendidas para programação de computadores em disciplinas profissionalizantes, a totalidade deles respondeu que os professores não utilizam esses conhecimentos. Às vezes, usam programas comerciais prontos e comparam alguns resultados analógicos (via manual) com os resultados dos mesmos. Deve-se assinalar a exceção observada nos cursos de engenharia elétrica, onde diversas disciplinas fazem uso das técnicas de programação para desenvolver tarefas específicas.

Os professores das disciplinas profissionalizantes não utilizam o computador como ferramenta para programar, apenas para executar programas comerciais. Em nenhum dos casos, seja na área de estruturas, hidráulica, gerenciamento, construção civil, medidas eletrônicas, mecânica dos sólidos, fabricação, usinagem, entre outras, são desenvolvidas aplicações específicas usando técnicas de programação.

No passado, quase todos os programas computacionais eram produzidos no ambiente da graduação e pós-graduação. A maioria dos alunos os desenvolvia facilmente, com programação seqüencial e o uso da linguagem Fortran. No presente, é difícil desenvolver softwares com interface gráfica moderna no ambiente universitário.

### **3. ENSINAR A PROGRAMAR OU USAR PROGRAMAS PRONTOS?**

Deve-se lembrar que nem tudo está pronto no mundo dos computadores e, muito menos, a solução a diversos problemas de engenharia está sendo pesquisada, para os quais os engenheiros irão programar novas soluções em novos modelos. Portanto, quem vai adaptar os atuais programas a futuras plataformas de computação (sistemas operacionais e hardwares) são os atuais profissionais, que, contudo, não foram preparados para isso, nem aqueles que estamos hoje preparando. Serão os atuais alunos capazes de se tornarem bons programadores se apenas aprendem a utilizar software e não têm capacitação adequada para essa atividade?

Seria muito fácil ensinar os alunos a utilizarem softwares prontos, pois basta comprar os programas e dar-lhes algumas explicações sobre como usá-los. Isso evitaria os comentários, corretos, que os alunos fazem em relação a aprender muito pouco sobre nada.

Poderia, também, ser considerada a criação de disciplinas optativas para a aprendizagem de programação formal de computadores, as quais, contudo, seriam pouco procuradas por requerem talentos e aptidões especiais. Talvez uma solução seja equilibrar o ensino de programação apenas usando planilhas de cálculo para resolver diferentes aplicações ao longo do curso, focando a importância nas boas práticas de raciocínio lógico-matemático que podem ser programadas.

A lógica computacional apresenta uma barreira importante para o bom andamento das disciplinas de ensino de programação, pois demanda a capacidade de serem definidos algoritmos seqüenciais ou orientados a objeto para caracterização dos processos computacionais básicos. A completa novidade desse conhecimento inibe o aluno, pois modifica sua maneira de pensar elaborada no curso de ensino médio e básico. BIANCHETTI (2001) mostra a dificuldade de se lidar com o conhecimento por simulação que caracteriza a cultura informática. Evidencia-se, assim, segundo o autor, uma contraposição interessante que demandaria novas investigações: há uma nova lógica indutiva, calcada na operação física, e uma outra dedutiva, atrelada à informática. Na primeira, aparentemente, o usuário pensa por que fez (programa computadores, ou faz a mão); na segunda, supostamente, pensa (usando programas prontos), mas não faz.

A tecnologia digital é uma caixa preta semi-aberta porque se pode chegar ao seu conteúdo só até certo ponto. Os profissionais, professores e alunos não querem entender que, quando se tem uma compreensão, uma tecnologia geral, um conhecimento teórico forte, pode-

se dominar a programação de computadores, pois reconhece se a lógica da seqüência de passos a seguir. Também se pode até ser crítico em relação a resultados de programas prontos.

#### **4. CONCLUSÕES E COMENTÁRIOS FINAIS**

Como nas diversas áreas abordadas num curso de engenharia existe um elevado número de aplicações disponíveis, é claramente irrealista ter-se a intenção de ensinar a totalidade dos alunos a utilizarem um grande número de programas. Nessas circunstâncias, correr-se-ia o risco de diminuir drasticamente o número de matérias tradicionais para abrir espaço no plano de estudos para o treino dos futuros utilizadores de software profissional. Não se deve também manter um tipo de ensino 100% tradicional, em que o contato dos alunos com os computadores seja demasiado escasso. Portanto, um equilíbrio entre essas duas situações parece ser a solução mais adequada.

Quanto à questão de desenvolver localmente aplicações ou de utilizar software profissional, deve-se adotar a estratégia de dividir as turmas em dois grupos: um primeiro, pouco numeroso, formado por aqueles com adequado talento e que recebam ao longo do curso constantes ensinamentos na área da programação, um outro grupo, mais numeroso, que tenha um reduzido contato com a programação e que se dedique essencialmente à utilização de software profissional. A divisão dos alunos nesses dois grupos basear-se-ia na freqüência de diferentes cursos extracurriculares, ou por intermédio de disciplinas de opção livre, que teriam de existir em quase todos os anos da graduação.

Uma alternativa válida é realizar teste de proficiência no uso de computadores, tal como várias universidades e cursos de engenharia vêm fazendo, e deixar para disciplinas de lógica de programação, nos semestres intermediários dos cursos, a ciência dos algoritmos, que pode ser desenvolvida usando planilhas de cálculo. Aprender a utilizar programas comerciais não deveria ser do âmbito do ensino profissional da engenharia, mas de cursos de extensão reduzidos, que até são mais econômicos que aqueles desenvolvidos durante um semestre ou ano letivo. A exceção apenas deve se dar nos cursos de engenharia eletrônica, onde deve ser obrigatório o ensino formal da programação lógica e física dos computadores e sistemas de dados.

#### **REFERENCIAS BIBLIOGRAFICAS**

BARROS, E.A.R, ZAMBONI, L.C., PAMBOUKIAN, L.C. Ensino de computação para estudantes de engenharia. In: XXXII CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA, 2004, Brasília. **Anais ...** Brasília: UNB, 2004.

BIANCHETTI, L. **Da chave de fenda ao laptop**. Petrópolis: Vozes, 2001.

FERREIRA, J.A. O ensino da ciência da computação nos cursos de graduação em engenharia da UFOP: necessidade de uma nova abordagem. In: XXVIII CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA, 2000, Ouro Preto. **Anais ...** Natal: UFOP, 2000.

FILHO, P.D.M. Uma experiência em ancorar o aprendizado de computação básica na construção de modelos computacionais de engenharia. In: XXIX CONGRESSO

BRASILEIRO DE ENSINO DE ENGENHARIA, 2001, Porto Alegre. **Anais ...** Porto Alegre: PUC-RS, 2001.

OLIVEIRA, C.A., ALVES, JB.M. Ciências da computação nos cursos de engenharia: uma proposta pedagógica inovadora. In: XXIX CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA, 2001, Porto Alegre. **Anais ...** Porto Alegre: PUC-RS, 2001.

PRAVIA, Z.M.C., KRIPKA, M. Proposta metodológica para desenvolvimento de ferramentas computacionais no ensino de estruturas. In: XXVII CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA, 1999, Natal. **Anais ...** Natal: UFRN, 1999.

## COMPUTATION IN THE ENGINEERING: TEACH PROGRAMING AND/OR USE COMMERCIAL SOFTWARE?

**Abstract:** *The computation teaching in the engineering courses this being obsolete and not supplying subsidies for the formation of the futures professionals. The recent proliferation of low cost software's, other not so much, in the teaching institutions, in the offices of studies and projects, it came to increase the pressure on educator and students in the sense of increasing the use of the new technologies in the academics atmosphere. While in a past recent, few engineers worked regularly with a computer, in the days today practically all recently-formed are at workstations with the most modern tools of the computation are use. Evaluating the curricula of several institutions and their programs for computation teaching systematically, as well as the challenges on what to teach: programming of computers and/or use of commercial software's, this work presents the current state of the computation teaching, and some alternatives that can be considered for the necessary level for the engineers that will be formed in the century XXI, in a market where the science of the information is everything.*

Keywords: Computation, Engineering, Teaching.