

INFORMÁTICA

83



**XVI CONGRESSO
NACIONAL DE
INFORMÁTICA**

SÃO PAULO OUTUBRO 83

ENSINO DE COMPUTAÇÃO EM COLÉGIOS

Alberto A.B. Furtado e Antonio L. Furtado

Rua Marquês de São Vicente, 225
22453 - Rio de Janeiro - RJ
Tel: 274-9922 R: 385/386
274-4449

Palavras chaves: ensino de computação, programação, linguagens de alto nível, resolução de problemas.

Resumo

A penetração do computador em atividades profissionais e mesmo na vida cotidiana requer uma preparação que, para atingir amplamente a população, deve iniciar-se nos colégios. O presente trabalho indica os objetivos a serem alcançados pelo ensino da computação em colégios, a forma de motivar os alunos e um possível programa de curso; oferece ainda sugestões quanto à metodologia de ensino e de avaliação de aprendizagem. O trabalho se baseia na experiência obtida com alunos de curso científico do Colégio Teresiano. Os alunos têm acesso ao equipamento de computação da Pontifícia Universidade Católica do Rio de Janeiro.

1. Introdução: Objetivos e Motivação

Um número crescente de colégios vem se interessando em incluir cursos de computação em seus currículos, dentro do programa de ensino profissionalizante ou como matéria optativa. Esta tendência reflete a penetração do computador em diversas atividades profissionais e mesmo na vida cotidiana.

Dois objetivos do ensino de computação podem ser identificados:

- a. compreender o que são e para que servem computadores;
- b. aprender a usar computadores.

O objetivo a não significa apenas mais um item de cultura geral. Todo indivíduo deve estar apto a refutar afirmações como: "esta decisão está certa (ou não pode ser alterada), porque foi tomada pelo computador", ou a afirmação oposta mas que é também um modo de as pessoas se eximirem de suas responsabilidades: "isso foi erro do computador". Ainda mais importante, é a preparação de usuários que saibam o que podem pedir de serviços de computação em suas futuras atividades profissionais.

Quanto ao objetivo b, pensa-se logo na formação de futuros especialistas em computação. No entanto, com o uso cada vez mais amplo de micro-computadores de uso pessoal, tende a aumentar a probabilidade de que um número crescente dos alunos terá acesso a esses equipamentos mesmo sem serem especialistas na área.

Um benefício indireto, ainda dentro do objetivo b, é a educação do raciocínio sistemático através do desenvolvimento de algoritmos e de sua implementação sob a forma de programas.

A explicação desses objetivos aos alunos no decorrer de um curso de computação ajuda a motivá-los. É essencial tirar também partido de um outro fator de motivação, bem adequado a alunos adolescentes, que é o computador como jogo ou como brinquedo. Para isso é indispensável que um bom número de programas dos alunos sejam efetivamente rodados. Se possível, os alunos devem ter a oportunidade não só de perfurar (ou digitar) seus programas mas também de operar o próprio computador.

Nas seções seguintes discutiremos, com base na experiência obtida com alunos do curso científico do Colégio Teresiano, um possível programa de estudos e apresentaremos indicações quanto à metodologia de ensino e de avaliação de aprendizagem.

2. O Conteúdo do Curso

A ênfase maior é em programação. Entretanto, tendo em vista os objetivos mencionados na seção anterior, são examinados os seguintes itens:

- a. Noções de "hardware"

- b. "Software" básico: sistemas operacionais e compiladores

- c. Aplicações de computadores

- d. As profissões relacionadas a computadores e situação de mercado de trabalho no Brasil.

- e. Situação da indústria de "hardware" e "software" no Brasil.

Na parte de aplicações, procura-se mostrar a diversidade de áreas em que se pode usar um computador. Deve-se mencionar o processamento numérico como apoio à engenharia, estatística, etc., o processamento não-numérico (gráfico e de textos, por exemplo), o processamento comercial e também a automação para controle de processos industriais.

Antes de entrar em programação, é introduzida a noção fundamental de

f. algoritmo

como método sistemático para resolução de problemas. Todos os exercícios de programação que se seguirão devem ser precedidos do projeto de um algoritmo adequado.

A parte de programação é dividida conforme relacionado abaixo:

g. programação:

- elementos de uma linguagem de alto nível;
- programas numéricos sequenciais;
- estruturas de controle;
- "arrays";
- programas gráficos;
- dados não numéricos: caracteres e valores lógicos;
- arquivos e programas comerciais;
- procedimentos e desenvolvimento modular de programas.

A ordem de apresentação dos assuntos a a g não é necessariamente a que relacionamos. Cada instrutor pode adaptar a seqüência dos assuntos a seu estilo de exposição e à maturidade e interesse dos alunos. A duração do programa é de um ano letivo.

Há muitas escolhas possíveis de linguagem de alto nível: Pascal, PL/I, FORTRAN, Basic, etc. As linguagens Pascal e Basic têm a vantagem de sua ampla disponibilidade em micro-computadores. A linguagem PL/I é usada para programação comercial em algumas instalações. A maioria das aplicações numéricas têm sido feitas em FORTRAN. Não aconselhamos COBOL para um primeiro curso, apesar de ser a linguagem mais usada para programação comercial, por ser de aprendizagem mais difícil.

Os alunos devem ser, bem cedo, habituados a escrever e rodar programas. Isto se torna possível já com o item de programas numéricos sequenciais (segundo item do tópico de programação). Trata-se de programas que simulam uma máquina de calcular: entrada de dados, alguns cálculos e saída de dados.

O item de estruturas de controle segue os princípios da programação estruturada. Nossa experiência sugere que as estruturas;

- se (condição) (comandos)
- senão (comandos)
- enquanto (condição)
- repita (comandos)

são de mais fácil aprendizagem que a estrutura:

- para (variável) desde (limite inferior) até (limite superior) repita (comandos)
- a qual é usada, em especial, na manipulação de "arrays".

Os programas gráficos, envolvendo desenhos e apelando para conhecimentos de geometria, são bastante motivantes. Temos adotado um pacote de sub-rotinas não muito poderosas mas fáceis de implementar e utilizar. É observada aos alunos a importância da programação gráfica para a área de CAD/CAM, de crescente interesse no desenvolvimento de projetos de engenharia.

3. Técnicas de ensino e avaliação de aprendizagem

A matéria do curso é apresentada aos alunos sob a forma de exemplos, visando estimular o aluno com material que ele possa utilizar no dia a dia da escola. Dependendo do ano em que o aluno esteja, diferentes exemplos devem ser utilizados.

Aconselhamos que este tipo de curso seja dado no terceiro ano científico para que o aluno melhor possa usar as ferramentas que lhe são dadas, por ter maior base matemática. Já que a maioria dos exemplos dados são de base matemática, essa base é essencial. Entretanto, como a colocação deste curso às vezes não é possível no terceiro ano por causa da intensa carga imposta pelos vestibulares, aconselha-se, pelo menos, que o curso seja retardado ao máximo.

As provas são dirigidas para saber se o aluno conseguiu reter as técnicas apresentadas através dos exemplos dados, e se consegue utilizar essas ferramentas para solucionar novos problemas. Destas provas 75% são perguntas

do tipo "faça um programa para resolver o seguinte problema". São dadas duas provas por trimestre. Antes da data de cada prova os alunos devem concluir três trabalhos práticos, sendo avisados de que os trabalhos constituem preparação para as provas.

4. Avaliação dos resultados e considerações finais

Os alunos em geral são bastante receptivos e interessados neste tipo de atividade, o que é comprovado pelas altas médias alcançadas pelas turmas (em torno de 8,0, com uma variância de aproximadamente 1,3). Esta média foi alcançada em provas de dificuldade entre média e alta.

Um cuidado especial deve ser tomado quando for dada a parte de "arrays" e problemas gráficos, especialmente quando se tratar de turmas de pouca base matemática e lógica. Estes tópicos costumam ser difíceis de assimilar pela maioria e deve-se evitar uma excessiva cobrança destes tópicos para não desencorajar os alunos.

O aproveitamento demonstrado nos trabalhos e provas e o grau de interesse revelado têm sido animadores. Deverão, entretanto, ser novamente avaliados após os alunos deixarem o colégio. Consideramos de especial interesse

acompanhar esses alunos durante o período universitário, verificando se existe correlação significativa entre esses estudos de computação e o posterior desempenho acadêmico.

Bibliografia

- (1) R.A. Barrett - Anais do 13th SIGCSE Technical Symposium on Computer Science Education - ACM/SIGCSE Bulletin, vol. 14, nº 1, 1982.
 - (2) K.L. Bowles - Problem solving using Pascal - Springer (1977).
 - (3) S.E.R. de Carvalho - Introdução à ciência da computação com WATFIV e FORTRAN - Campus (1981).
 - (4) A.L. Furtado, A.A.B. Furtado e F.A. Messeder - Instructional graphics packages to be used with a line printer - ACM/SIGCSE Bulletin, vol. 14, nº 4, (1982).
 - (5) A.L. Furtado e E.P.L. Passos - Introdução à programação com PL/I - Livros Técnicos e Científicos (1976).
 - (6) R.C. Holt e J.N.P. Hume - Fundamentals of structured programming using FORTRAN with SF/K and WATFIV-S-Reston (1977).
 - (7) S. Papert - Uses of technology to enhance education - AI - M - 298, The Artificial Intelligence Lab., MIT (1973).
-