

***** RELATÓRIO TÉCNICO *****
EXPERIÊNCIA DO ENSINO DE
COMPUTAÇÃO I - UM ESTUDO DE CASO

Eduardo Peixoto Paz
José Antônio dos Santos Borges

NCE 24/91
Dezembro/91

Universidade Federal do Rio de Janeiro
Núcleo de Computação Eletrônica
Caixa Postal 2324
20001 - Rio de Janeiro - RJ
BRASIL

Este artigo foi apresentado no I Congresso Iberoamericano de Educação Superior em Computação, realizado em Santiago do Chile, de 14 a 15 de outubro de 1991.

EXPERIÊNCIA DO ENSINO DE COMPUTAÇÃO I - UM ESTUDO DE CASO

RESUMO

Este trabalho descreve o processo de automatização do Curso Básico de Computação da Universidade Federal do Rio de Janeiro. É apresentado o sistema AMBIENTE, que executa sob UNIX e foi concebido para ser uma interface amigável com os estudantes e professores desse curso. São também descritos os problemas que conduziram às soluções que o AMBIENTE proporciona, tanto do ponto de vista operacional quanto de manejo de pessoal.

EXPERIENCE ON TEACHING A COMPUTER PROGRAMMING COURSE - A CASE STUDY

ABSTRACT

This paper describes the automatization process of the Introductory Course of Computer Programming at the Federal University of Rio de Janeiro. The system, called "AMBIENTE" runs under Unix operating system and was designed to have a friendly interface with teachers and students. The problems which led to the solutions used in the "AMBIENTE" system are also described, both from an operational and personal utilization points of view.

1) Histórico

A Universidade Federal do Rio de Janeiro oferece para seus alunos o curso de introdução ao uso de computadores (Computação I). Este curso teve início no Departamento de Ciência da Computação, no Instituto de Matemática, que ficou responsabilizado em oferecer aos demais departamentos o curso básico de computação. Com a disseminação do uso de computadores, nas mais diversas áreas, um grande número de alunos passou a necessitar desta disciplina. Hoje a oferta de vagas atende a mil alunos por semestre.

O crescimento do número de alunos em um período curto de tempo, levou a organização do curso por diferentes caminhos, buscando melhorar o aprendizado dos alunos e a gerencia do curso. Com um grupo bastante heterogêneo de alunos e uma abordagem de ensino livre, ocorria um alto índice de desistência do curso assim como a redução no grau de exigência quanto ao conteúdo programático. Tanto alunos quanto professores ficavam desestimulados.

Para suporte prático ao curso, são usados os recursos computacionais do centro de processamento de dados da Universidade - Núcleo de Computação Eletrônica. Um curso com tantos alunos e professores necessita de um suporte de laboratório bastante eficaz e de uma gerencia bem próxima à sua dinâmica diária.

2) Características do pessoal envolvido

Para melhor compreensão do trabalho, daremos as características básicas dos diversos grupos de pessoas envolvidas com o curso de Computação I. São atores deste cenário: os alunos, os professores, o departamento e o centro de processamento de dados.

a) Alunos

O conhecimento dos alunos que procuram o curso é bastante heterogêneo. Existem alunos que estão ingressando na universidade, alunos por se formar, cursando, em especial, engenharia, química, astronomia, física e geologia. De um modo geral podemos dividir os alunos em um grupo com e outro sem dificuldades no aprendizado da computação.

Os alunos com dificuldades geralmente estão entrando para a universidade. Estes alunos sofrem o processo de mudança no estilo do ensino, do curso secundário para o universitário. Alguns alunos não têm sequer prática no uso de um teclado de máquina de escrever, outros acreditam que seus conhecimentos de matemática são insuficientes para aprender computação. São alunos que não têm o hábito de usar máquinas na sua vida cotidiana e tem uma visão de computador de duas décadas atrás.

Os alunos sem dificuldades são aqueles que têm contato frequente com equipamentos eletrônicos tais como computadores pessoais, "video game", caixa eletrônico de bancos, etc. Para os alunos com conhecimento prévio de computação, normalmente o curso básico é insuficiente, estando sempre procurando mais e mais opções de uso (em alguns casos até atrapalhando os colegas).

b) Professores

Os professores estão ligados de alguma forma ao conhecimento de computação. Não é exigido que o professor seja um programador experiente. Muitos dos professores não possuem conhecimentos completos sobre a linguagem e muito menos sobre o sistema operacional da máquina usada para elaboração dos exercícios práticos. Como os sistemas operacionais são bastante amplos, com poucas aulas encontramos alunos com conhecimentos mais aprofundados que o professor sobre assuntos específicos e procurando o conhecimento de mais detalhes. Este fato afasta o professor do laboratório e o faz enfatizar o ensino da linguagem em detrimento do ensino de programação.

c) Departamento

O departamento responsável pelo ensino de computação aplica duas versões do curso: uma específica para os alunos de informática (aproximadamente 40 vagas) e outra para alunos em geral (aproximadamente 1000 vagas). Dentro do departamento, o coordenador da cadeira tem autonomia para implantar todas as mudanças necessárias para o bom funcionamento do curso. Existe uma dificuldade básica de avaliação se estas mudanças são realmente positivas pois os mecanismos de realimentação são tenues.

d) Centro de processamento de dados (NCE).

A parte prática do curso é realizada no NCE em uma máquina que necessita apoio operacional. Para o funcionamento do curso o apoio deve ser constante. A equipe do NCE encarregada desta tarefa está acostumada a fornecer consultoria de alto nível em implantação de sistemas não possuindo pessoal para atender uma grande massa de alunos com solicitações constantes. Para o atendimento ao aluno, é preciso o conhecimento do desenrolar do curso para avaliar suas solicitações, e dessa forma, o atendimento ao aluno principiante, por parte do NCE é muito pouco efetivo.

3) A operação do curso de Computação I ao longo dos anos

Nas primeiras vezes em que o curso de Computação I foi aplicado, o ensino seguiu uma linha tradicional, descentralizada, onde cada professor era responsável pela sua turma. Cada professor possuía métodos diferentes de avaliação e didática fazendo com que alunos com mesmo diploma tivessem conhecimentos diferentes. Isso também trazia alguns problemas quando os alunos passavam para disciplinas mais avançadas, que assumiam que o aluno houvesse cursado Computação I.

Uma sequência natural foi a centralização, com a criação da figura de um coordenador, aplicação de prova única, preparação de exercícios padronizados, utilização de um único tipo de computador, etc. A centralização visava principalmente: a redução do trabalho do professor, avaliação normalizada e coletiva, além de procurar otimizar o uso de laboratórios (terminais, microcomputadores, etc...).

A disponibilidade de microcomputadores em ampla escala, nos levou a utilizá-los em Computação I, esperando que os problemas

operacionais (disponibilidade de equipamento, simplicidade de operação, etc...) melhorassem a qualidade do curso. Nossa expectativa foi frustrada, pois a causa real dos problemas não é o tipo de equipamento que se usa.

Para resolver uma série de problemas operacionais, voltou-se ao sistema centralizado, sob UNIX executando num VAX 780 dedicado. Foi criado um sistema automático para acompanhamento dos alunos, visando especialmente o controle da entrega de trabalhos e consulta de notas.

O sistema foi definido e programado, mas nunca conseguiu funcionar direito. Foi definido como uma coisa simples mas, como "em computação tudo se pode fazer" foi crescendo de forma tão desordenada, que em 6 meses virou um sistema monstro, que não conseguia mais servir como sistema de entrega, dada a necessidade de controle operacional muito dedicado. Os módulos mais necessários, a gerência de notas e o controle de presença não funcionaram por problemas de falta de pessoal para operar o sistema.

Aliado a isso, o rendimento dos alunos principiantes usando UNIX foi baixíssimo, pois este sistema não é preparado para ser operado por gente que nunca trabalhou com computação: tem uma linguagem de controle horrível, um sistema de arquivos muito elegante mas difícil de ser manipulado, e mensagens nada amigáveis. Os professores, que nunca haviam trabalhado com UNIX, passaram aos monitores a tarefa de explicar detalhes operacionais para os monitores, que se tornaram excelentes operadores de SHELL (linguagem de comando UNIX) mas péssimos programadores PASCAL...

4 Soluções com automação experiências e dificuldades

Uma constatação importante é a instabilidade das soluções apresentadas. Uma olhada no passado mostra que a evolução tecnológica e problemas operacionais empurram com muita rapidez, soluções brilhantes de organização para a desordem e o descontrole. Nossa experiência aponta para uma duração média de tres anos por solução. O declínio da qualidade do ensino é detetado, soluções são apresentadas, aumento da qualidade seguido de declínio. O processo é ciclico e tem basicamente o seguinte desenrolar.

Dado um cenário com problemas diversos, um professor é indicado para apresentar soluções. Para o cenário existente são propostas reformas objetivando solucionar os problemas. Geralmente a implementação destas reformas envolve um grande número de pessoas, principalmente considerando que sempre os alunos são envolvidos. Os problemas existentes são eliminados ou reduzidos, porém surgem novas situações. O professor, já incomodado com a tarefa da reforma inicial começa a receber novas demandas de mudanças com um novo cenário. Propostas complementares são formuladas para problemas muitas vezes não muito claros. O professor perde o objetivo inicial da reforma. Este é o início da degradação da solução. É importante se ter a consciencia que o cenário é dinâmico e precisa de avaliações profundas e periódicas. Este é o momento crítico do processo, momento onde todos são induzidos a encontrar outro professor responsável por novas mudanças e o ciclo é reiniciado.

A experiência adquirida na gerencia do curso é muito difícil de ser transmitida para outro professor pois possui uma enorme componente subjetiva. Em muitos casos é mais simples jogar fora todo o trabalho anterior do que remendar uma solução cheia de emendas.

Como exemplos de degradação temos a tecnológica e a operacional. A tecnológica está ligada ao uso de novos equipamentos, sistemas operacionais, linguagens de programação. Este tipo de demanda vem tanto dos alunos e departamentos envolvidos como dos próprios professores da disciplina. A degradação operacional vem principalmente do fato do professor ficar encarregado de uma tarefa administrativa bastante intensa. São propostas constantes de pequenas mudanças operacionais, mudanças de metodologia de ensino, atualizações de notas, exercícios, provas etc. É muito fácil trocar-se objetivos gerais e duradouros por específicos e de curto prazo.

5) Pré-requisitos de uma solução

Todos os erros que foram cometidos na implementação e implantação do sistema automático que foi descrito nos levaram a repensar o sistema, agora de forma mais global. Nossa idéia foi tentar encontrar os elementos que seriam indispensáveis atacar para conseguir uma solução. Os itens a seguir foram aqueles que julgamos "chave".

a) Aluno tem que estar sempre bem informado

Num curso que tem quase 1000 alunos, provindos de dezenas de departamentos diferentes, com nível de conhecimento quase zero do funcionamento da universidade (são, quase sempre, alunos de 1º ano), perguntas simples como p. ex. "qual é o dia de entrega do trabalho?" podem ser razão até mesmo para a reprovação de um aluno.

Por outro lado, um coordenador ou secretária não pode dispor de tempo para atender a essa massa de alunos, ou seja, é o professor tem que responder a qualquer pergunta, técnica ou organizacional, e portanto também estar completamente a par de tudo que se passa.

b) O elemento chave do processo de ensino é o professor

Nunca é demais dizer que existem professores melhores e piores, com maior ou menor interesse, cujas turmas tem melhor ou pior desempenho (historicamente falando). Diferenças individuais à parte, quase sempre a dedicação e o prazer do professor ao dar o curso são fundamentais para o sucesso do ensino.

A solução para o problema de ensino, então envolve conseguir que o professor se dê ao máximo ao curso, num envolvimento de intenso prazer, de corpo e alma. Tem que estar no laboratório, acompanhando de perto o trabalho do aluno e achar essa tarefa mui-

to agradável (Utopia ?)

c) Aluno não pode ter medo do curso

Computação, especialmente para quem não teve contato anterior com computador, é uma coisa amedrontadora. Para fazer um mínimo de tarefas, é necessário utilizar conceitos abstratos, decorar comandos, dialogar com um ser invisível (sistema operacional) que admoesta o usuário em caso de erro.

Assim, é necessário, para fazer com que o aluno consiga perder rapidamente o medo do computador, que o diálogo "homem-máquina" seja tão natural quanto possível. Entre as principais características de uma interface com usuário (que deve cobrir completamente o sistema operacional da máquina) estão os seguintes elementos:

- . sistema deve ter uma linguagem de comandos direta e intuitiva
- . sistema deve usar a língua do aluno (portugues, e não inglês)
- . sistema deve perguntar de forma muito simples (p. ex. menus)
- . aluno deve responder de forma muito simples (p. ex. com uma tecla)

d) Edição de textos

A edição de textos, fundamental num curso de Computação I, deve ser muito simplificada: computador deve se comportar exatamente igual a máquina de escrever. O editor deve ser "estilo full-screen", tipo WYSWYG, mas os comandos devem ser em menor número possível (se possível não usar controls, ALTS, etc.), possibilitando a aprendizagem em curtíssimo espaço de tempo. Como os textos (programas) criados são pequenos o editor não precisa ser poderoso. É importante notar que quanto antes o aluno consiga teclar, tanto mais rápido poderá treinar a parte de programação, que é o que nos interessa.

e) O aluno deve ser induzido a fazer muitos exercícios

Uma maneira de conseguir isso é apresentar muitos exemplos já teclados e induzir o aluno a modificá-los. Deve-se pensar que teclar um programa consome tempo, e economizando tempo de teclagem, o aluno terá mais tempo para entender a parte conceitual e fazer experiências.

f) A gerência de todo sistema deve ser centralizada

Distribuição de exercícios, comunicações, submissão de trabalhos, controle de notas, etc. devem ser controladas de maneira centralizada para garantir a disponibilidade e controle das informações, que são a alma do curso. O uso de um computador único para suportar o sistema, facilita bastante essa tarefa.

g) A solução automatizada deve ser simples

É importante notar que soluções completamente automatizadas para um problema tão "fluido" como controle acadêmico, com milha-

res de exceções são enormes. Assim, a solução deve ser bem definida para automatizar completamente o indispensável, e automatizar parcialmente ou não automatizar o que for mutável ou complicado. O sistema todo deve ser operável por uma única pessoa, se possível.

Essa simplicidade de solução visa garantir a flexibilidade de operação, mudanças mais rápidas no sistema, e até mesmo atualização manual nos arquivos se for necessário, para resolver casos não previstos.

Por outro lado estamos tratando com "terríveis" alunos, que disputam entre si maneiras de fraudar ou derrubar o sistema. O sistema deve ser muito robusto.

6 Implementação do sistema

A solução que foi adotada gira em torno de um pacote de software (chamado AMBIENTE) que atua como um "Shell" do sistema operacional UNIX. O sistema executa em um VAX-780, dedicado, com 64 terminais no máximo, com ULTRIX (Digital).

Ao início do curso os alunos recebem uma conta e uma senha. Ao entrar com eles num terminal, o AMBIENTE é ativado automaticamente, tomando conta do terminal e escondendo totalmente o sistema UNIX. Na tela aparece um "jornal" com as últimas novidades para que o aluno esteja sempre informado sobre o curso e a universidade (figuras 1, 2).

As opções que o AMBIENTE oferece são todas apresentadas como menus, que o aluno seleciona através de letras (figura 3). São poucos menus e todos em português. Os menus são apresentados, opcionalmente na forma abreviada para economizar tempo de entrada e saída.

O elemento mais usado no sistema é o editor de tela (figuras 4, 5), em que o aluno tecla seus programas ou dados. A forma de operação é muito simples: o usuário tecla e isso sai escrito na tela. Existem apenas os seguintes comandos, controlados por uma tecla: apaga letra sob o cursor, passa para a próxima linha e fim. Existe um menu auxiliar que permite ainda inserir e remover linhas, e auto-teclar um programa protótipo Pascal ou um cabeçalho de programa. Ao sair do editor (ESCAPE) o texto é sempre salvo (o que evita que os alunos percam inadvertidamente o que estavam teclando).

Uma vez teclado o programa, o usuário manda executar apertando uma tecla (figura 5). O processo de compilação e link-edição fica escondido do usuário, mas se houver erros (de compilação ou execução), o editor é automaticamente chamado com o cursor sobre o ponto de erro. É exercido um controle muito rígido sobre a execução e erros, garantindo que somente programas perfeitos executem (devemos lembrar que os alunos são principiantes e é melhor ser rigoroso).

Obs na UFRJ, em Computação I usa-se o Berkeley PASCAL Interpreter

ter

Uma das principais dificuldades que ao longo dos anos notamos que todos os alunos tem é o entendimento do sistema de arquivos, que, no caso do UNIX é especialmente complexo. Resolvemos que a manipulação dos arquivos seria feita de maneira totalmente interativa e intuitiva.

Para isso foi construído um manipulador de arquivos (figura 6) que mostra na tela o conteúdo do diretório atual e permite movendo o cursor e apertando uma tecla (indicada por um menu) apagar, listar, imprimir, duplicar, renomear, etc. Foram tornados acessíveis ao aluno apenas 2 diretórios: o diretório do professor, que contém os programas exemplo e o diretório do aluno, em que ele realiza suas operações. Não permitimos o caminhar na árvore do UNIX, o que diminuiu sensivelmente a necessidade de controle de proteção de arquivos.

Foi também criado um sistema de troca de mensagens que permite aos alunos trocarem impressões e sugestões (eventualmente ofensas). Esse sistema foi limitado para não permitir troca de arquivos o que obviamente seria usado para troca de soluções de programas/exercícios.

Por último foi criado um sistema de controle de notas e submissão de trabalhos, que permite a administração impessoal de prazos e informação atualizada de notas. Os trabalhos são submetidos pelos alunos e enviados ao monitor da turma para correção. As notas são digitadas e enviadas ao coordenador que realiza a atualização de um cadastro, que pode ser consultado pelos alunos facilmente. (figura 7).

O sistema tem algumas soluções originais, tais como o processo de cadastramento, em que é o próprio aluno que informa seus dados, pelo computador, teclando um texto "livre", como um exercício do curso. O interessante é que o trabalho de consistência é mínimo, dada a condição de que o aluno se sente responsável por seus próprios dados.

7) Papéis no sistema

Na definição do sistema ficou muito claro que é absolutamente indispensável a definição dos papéis das pessoas envolvidas: aluno, professor, monitor e coordenador. Esses papéis são monitorados pelo sistema computadorizado, tornando claro, sem regras explícitas e regulamentos (ou seja de forma intuitiva, se possível) seus limites e seus deveres.

Essa maneira de abordar o problema dá uma segurança muito grande aos participantes do sistema, sem que haja um cerceamento de poderes explícito (que na verdade, existe). Neste contexto são os seguintes os papéis:

- a a figura do aluno
 - faz exercícios
 - faz trabalhos para nota

- b) a figura do professor
 - . dá aula teórica
 - . orienta alunos nas soluções dos exercícios
 - . determina prazos
 - . provas e trabalhos são unificados.
 - . cada professor é responsável pela preparação de apenas um dos trabalhos práticos.
- c) a figura do monitor
 - . auxilia o professor na aula prática
 - . corrige os trabalhos
 - . lança notas dos trabalhos
- d) a figura do coordenador
 - . cadastra alunos
 - . envia avisos coletivos
 - . cadastra exercícios
 - . coordena tudo

Costurando todos esses papéis está o AMBIENTE, que provê as facilidades de informação e interrelacionamento entre os participantes, que são cerca de 1000 pessoas que não se conhecem (ou se conhecem pouco).

É interessante notar que todos os elementos envolvidos utilizam o AMBIENTE e não o Shell nativo UNIX. Isso traz diversas consequências, entre outras, uma homogeneidade de linguagem entre os participantes (ou seja não é necessário fazer treinamentos múltiplos entre os usuários). O coordenador é a única pessoa que precisa, em circunstâncias especiais, sair do AMBIENTE, nas situações que exigem privilégios especiais, tais como instalação do sistema, modificações não corriqueiras no cadastro ou o lançamento de notas.

8) Avaliação do AMBIENTE

O sistema AMBIENTE foi implantado em 1991. Resultados preliminares já demonstram resultados muito promissores, entre eles:

- diminuição a menos de metade do índice de desistências da disciplina.
- professores motivados
- alunos motivados, inclusive utilizando o AMBIENTE fora do seu horário de aulas, para treinar.
- alunos iniciantes sem receio da máquina
- turmas que tradicionalmente rejeitavam o ensino de computação, motivadas.
- Departamentos pedindo a inclusão da disciplina de Computação I como opcional em outros cursos.

O mais importante, entretanto é avaliar o incremento da qualidade de ensino. Para isso seria necessária uma avaliação comparativa muito cuidadosa, que não foi ainda realizada, pois o sistema foi apenas aplicado em 1 semestre, e não temos ainda dados que permitam uma análise estatística adequada.

Entretanto, podemos fazer uma comparação informal, tirada das opiniões dos 15 professores que em 1991 lecionaram a disciplina.

Já na primeira prova, os professores ficaram admirados com as notas obtidas pelos alunos, atribuindo os resultados a simplicidade da prova. Uma análise mais cuidadosa demonstrou que o nível da prova foi similar às provas dos semestres anteriores, com mesmo conteúdo programático, porém com resultados bastante melhores.

O sistema AMBIENTE foi implementado da forma mais simples possível, usando o mínimo possível de recursos especiais do UNIX. O AMBIENTE foi fundamentalmente programado por 2 pessoas, que cuidaram respectivamente da parte operacional do programa e das interfaces com o sistema operacional UNIX. A parte central do sistema, responsável por todo diálogo homem-máquina, edição de textos, controle de arquivos, ativação do compilador, monitorador de execução e troca de mensagens, tem aproximadamente 3500 linhas em linguagem C, incluindo comentários. O sistema de cadastramento de alunos e lançamento de notas, cerca de 2000 linhas de C e Pascal.

9) Conclusões

Não se deve imaginar que é fácil implantar (e impor) uma nova solução completa, ainda mais para um problema que tem 20 anos de existência. Durante a fase de implantação, o grupo de desenvolvimento sofreu todo tipo de pressões e desestímulos, tanto de alunos, monitores, professores e do departamento de Ciência da Computação. Todos achavam que não ia dar certo.

Isso é facilmente explicável. Um sistema de ensino pertence a uma categoria de problemas tipo "malha aberta": nunca se pode precisar exatamente os resultados, portanto mudanças erradas são feitas impunemente. E todos pagam pelo erro. Como avaliar se uma solução foi boa? Só o tempo responde.

O sistema foi implantado há 4 meses e, para nosso alívio, temos a comunidade satisfeita com os resultados. Há um consenso que, pelo menos até esse momento, houve realmente melhoria da qualidade de ensino, motivação dos professores, alunos e monitores e melhoria da organização.

Para concluir, todo o sistema AMBIENTE, incluindo os programas fonte e bibliotecas de exemplo que são usados no curso de Computação I da Universidade Federal do Rio de Janeiro, estão disponíveis, sem qualquer ônus, para qualquer instituição de ensino que deseje ter acesso a eles.

BIBLIOGRAFIA

1. Paz, E.P., Borges, J.A. e Stelling, R. - Documentação de operação do sistema AMBIENTE, Doc. Interna do NCE - 1991
2. Digital Corp, ULTRIX - Operating System Manual - 1989

