



1290000317



FE

TCC/UNICAMP B862s

João Sebastião de Oliveira Bueno

Software Livre Na Educação: A Integração da Informática ao Currículo

T B T H E E M C

Campinas

2003

UNICAMP - FE - BIBLIOTECA

João Sebastião de Oliveira Bueno

Software Livre Na Educação: A Integração da Informática ao Currículo

Trabalho de Conclusão de Curso

Orientador:

Sérgio Ferreira do Amaral

FACULDADE DE EDUCAÇÃO - UNICAMP

Campinas

2003

UNIDADE	FE
Nº CHAMADA:	ICC-UNICAMP
	33625
VOLUME	317
TÓPICO	124/2003
PROG.	
C. D.	X
PREÇO	11,00
DATA	05.11.03
Nº CHEQUE	Bib=308023

**Catálogo na Publicação elaborada pela biblioteca
da Faculdade de Educação/UNICAMP**

Bibliotecário: Gildenir Carolino Santos - CRB-8º/5447

Bueno, João Sebastião de Oliveira.

B862s Software livre na educação : a integração da informática ao currículo /
João Sebastião de Oliveira Bueno. -- Campinas, SP: [s.n.], 2003.

Orientador : Sérgio Ferreira do Amaral.

Trabalho de conclusão de curso (graduação) – Universidade Estadual
de Campinas, Faculdade de Educação.

1. Currículos. 2. Informática. 3. Softwares. 4. Escrita. 5.*Produção digital. I. Amaral, Sérgio Ferreira do. II. Universidade Estadual de Campinas. Faculdade de Educação. III. Título.

02-101-BFE

João Sebastião de Oliveira Bueno

**Software Livre Na Educação: A
Integração da Informática ao Currículo**

Campinas

2003

João Sebastião de Oliveira Bueno

Software Livre Na Educação: A Integração da Informática ao Currículo

Trabalho de Conclusão de Curso

Orientador:
Sérgio Ferreira do Amaral

FACULDADE DE EDUCAÇÃO - UNICAMP

Campinas

2003

Trabalho de Conclusão de Curso sob o título *Software Livre Na Educação: A Integração da Informática ao Currículo*, por João Sebastião de Oliveira Bueno

Sérgio Ferreira do Amaral
Orientador

Dirceu da Silva
Segundo Leitor

Resumo

Este trabalho coloca questões sobre a necessidade de se incorporar a informática de forma integral ao currículo escolar. Apresenta os modelos de desenvolvimento de *software* livre e proprietário e argumenta que o modelo proprietário não constitui uma alternativa viável para a educação. Expõe o impacto da computação pessoal na escrita formal até o ano corrente, e sugestões para a integração da informática ao ensino de várias disciplinas escolares. Apresenta um modelo de programação de computadores e propõe sua inclusão curricular. Conclui que a Educação não pode ignorar, nos dias atuais, a informática integrada ao ensino.

Abstract

This work puts out questions about the need of incorporating informatics into the school curriculum. It presents the models os proprietary and free software development, and suggests that the proprietary model is not a viable option for use in education. It exposes the impact of the Personal Computer in formal writting to the present date, and presents sugestions for informatics integration in the teaching of a number of school subjects. A model of computer programing is presented, and its inclusion in the curriculum is proposed. It concludes that education can not, at present, ignore informatics integrated to teaching.

Sumário

1	Educação e Informática	p. 1
1.1	Introdução	p. 1
1.2	Contexto Atual da Computação Pessoal	p. 6
2	Liberdade e Corporatividade	p. 9
2.1	A História da Incrível Máquina de Construir Casas	p. 9
2.2	Propriedade Intelectual	p. 10
2.3	Patentes de Software e Algumas Implicações	p. 15
2.4	Direitos Autorais e seus Prazos	p. 16
2.5	A Lei Além da Justiça: Proteção Contra Cópia	p. 18
2.6	Formalizando o Software Livre	p. 23
3	Escrevendo e Publicando	p. 31
3.1	A Escrita no Computador	p. 31
3.2	A Escrita Eletrônica como Produção Pessoal do Aluno	p. 35
3.3	A WEB	p. 37
4	O que já Existe para o Currículo Hoje	p. 48
4.1	Introdução	p. 48
4.2	Matemática	p. 49
4.3	Biologia	p. 53

4.4	Geografia	p. 54
4.5	Língua Estrangeira	p. 55
4.6	Química	p. 56
4.7	Física	p. 57
4.8	Fotografia e Artes Plásticas	p. 58
4.9	Música e som	p. 63
4.10	Projetos - Desenho Técnico	p. 64
4.11	Vídeo	p. 65
4.12	Finalizando	p. 67
5	Programação como Ferramenta de Criação	p. 68
5.1	Programar - O Que É?	p. 68
5.2	Dados Digitais	p. 71
5.3	O Funcionamento do Computador	p. 74
5.4	Dois Exemplos de Programa	p. 79
6	Conclusão	p. 88
	Referências Bibliográficas	p. 92

1 Educação e Informática

1.1 Introdução

Ao educador tradicional, uma pergunta que pode ocorrer é “por que me preocupar com o ensino de informática? Não é apenas mais uma entre tantas áreas do conhecimento, já abordadas pelas disciplinas escolares?”.

Neste capítulo, argumenta-se que a resposta a esta pergunta é não.

Começando por apenas um aspecto da informática, podemos avaliar que se por um lado a informática enquanto objeto de ciência está na infância, se comparada com as outras disciplinas ou saberes que constam do currículo escolar, por outro lado ela vem modificar profundamente a relação que se tem com a escrita, e em menor grau com a leitura. Ora, é sabido que a escrita enquanto linguagem é a base principal de transmissão dos saberes e conhecimentos de nossa sociedade. Antes de se popularizar, na Grécia clássica, por exemplo, era costume que obras fossem memorizadas e recontadas oralmente. O próprio Platão considerava o livro um péssimo professor¹. Com o advento da escrita o conhecimento formal teve alterada sua principal forma senão de transmissão, de conservação.

¹Paul Tannery in Platão, 1999, p.17

E é justamente a escrita que ao longo dos últimos dez - e apenas dez - anos se transformou de forma radical com a popularização dos computadores pessoais. As facilidades proporcionadas para introduzir modificações, efetuar correções, definir o layout, produzir cópias e impressões, transmitir o texto para lugares distantes em tempo praticamente zero, adequar um dado texto ao formato usado por outros, para construir uma coleção, e assim por diante fizeram com que em situações formais o manuscrito praticamente desaparecesse - e estamos falando de uma janela de não mais de dez anos para a maioria das situações.

Se o manuscrito se foi - e o datilografado mais ainda - então o que se usa para escrever? Os computadores, e dentro desses, um programa para editar textos. E para ler? Nesse ponto a informática ainda não introduziu modificações tão abrangentes, e a forma do texto utilizada para leitura é predominantemente a impressa em papel. Isso se dá por questões técnicas - ainda não há um dispositivo comercial que nem de longe propicie a comodidade, o conforto, a mobilidade, e sobretudo, a qualidade de um texto impresso em papel.

No entanto não há razões para crer que isso não se modifique também nos próximos dez ou vinte anos. Anúncios de tecnologias em desenvolvimento como o "papel eletrônico"² levam a crer o contrário. Por outro lado há também modificações na leitura. Sobretudo na leitura de informações mais voláteis como notícias e mesmo correspondência, ou resumos de obras. E aí, com a introdução do hipertexto na sua forma mais conhecida, a "World Wide Web", modifica as relações de leitura sobretudo no que diz respeito a linearidade, e tempo de acesso a assuntos relacionados.

²<http://www2.parc.com/dhl/projects/gyricon/>, acessado em 10/4/2003

Se a relação com a leitura e escrita a partir da informática se modifica tão fundamentalmente, deve-se mencionar que o computador pessoal não se restringe a uma máquina de escrever. Enquanto que sua relação perceptível a primeira vista com o ensino possa ser essa, tudo o que se relaciona a imagens e sons pode ser representado - e editado, alterado, revisto, utilizando-se os mesmos como ferramenta. Isso é válido inclusive para músicas, partituras, ilustrações, e filmes.

E, enquanto esses outros aspectos escaparem aos educadores, escapam também à maioria das pessoas. Ora, em nossa sociedade, as inovações tecnológicas tem se dado principalmente pela "demanda do mercado". Isso é, o foco de produção da tecnologia para uso final - para consumo - está nas empresas, e a essas interessa desenvolver o que lhes dará maior retorno financeiro.

Desta forma, ao mesmo tempo em que o computador pessoal nasce e amadurece com o potencial para se tornar o escritório - entendido como o lugar onde se escreve -, a biblioteca, o ateliê de pintura, o estúdio de cinema, a oficina de música, o videogame, e a oficina de programação - o potencial para que as próprias pessoas modifiquem e criem suas próprias ferramentas para as tarefas acima -, algumas dessas portas vão sendo gradualmente fechadas, de forma sutil, quase imperceptível, justamente por que não se desconhecem, esses potenciais. De todas essas possibilidades restam algumas poucas como a de escritório, biblioteca, e videogame, e principalmente central de compras, que ainda são amplamente divulgadas pelas forças da indústria de computação.

De fato um computador pessoal como os vendidos nos dias de hoje - 2003 - tem potencial para todas essas coisas sem quaisquer componentes físicos adicionais - isso é, sem mais componentes eletrônicos, circuitos, ou periféricos. Tudo isso pode ser

alcançado exclusivamente pelo uso de programas específicos.

O que ocorre é que ou se pensa a priori que tais programas seriam difíceis de se usar, ou se assume que sejam caríssimos, mais caros do que o computador em si.

Mas ainda que abordemos o computador pessoal apenas como instrumento de escrita, seu relacionamento com a educação seria já indissociável, e não necessariamente abordado numa disciplina separada de “informática”.

Não se apregoa aqui de forma alguma o fim da escrita a mão, com uso de lápis ou canetas - só se aponta para o fato de que hoje em dia, as situações formais de escrita exigem que esta se apresente sob a forma digital. No entanto, desenvolvimentos na área de interfaces de produtos de consumo de informática indicam que logo mesmo a escrita digital se dará como a fazemos sobre papel - com um instrumento parecido com uma caneta sobre uma superfície sensível, e uso de escrita cursiva.

Atualmente - como nos últimos anos - a forma mais comum de se dar entrada a um texto é através de um teclado, com layout herdado das máquinas de escrever, e um monitor do tipo CRT (Catode Ray Tube - a mesma tecnologia usada nas TVs), para visualização do texto digitado. Mais ainda, o conjunto de programas utilizado pela maioria das pessoas para digitar textos compõe-se do sistema operacional Microsoft Windows, e do processador de textos Microsoft Word.

Esse conjunto de ferramentas permite que além da digitação dos textos estes sejam formatados diretamente para o formato que se deseja, escolha de tipos e tamanhos de fontes diferente para cada parte do texto, e apresentação na tela no formato em que a mesma será impressa. O texto assim escrito pode ser salvo como

um arquivo - um objeto composto de dados digitais, que tem uma representação através da interface do sistema operacional - esse arquivo então pode ser guardado para leitura e modificação posteriores, impresso em uma impressora, enviado pela internet para outra pessoa, entre outras coisas.

Para algumas pessoas - educadores inclusive - parecerá que então o aprendizado de informática poderia ser reduzido ao ensino da operação do processador de texto, e manipulação do arquivo resultante, com um conjunto fixo de programas como o descrito acima.

Sendo um computador pessoal uma máquina capaz das mais diversas funcionalidades, como já foi dito, não é apropriado que se pense que seu uso estaria completo apenas se sabendo tão pouco sobre os mesmos. Por outro lado, devido a uma série de fatores, desde culturais, até imposições de um mercado que percebe qual imagem do computador poderia tornar as vendas mais fáceis, há uma tendência para esse tipo de pensar sobre a informática.

E esta visão é justamente a de uma caixa mágica, que faz tudo sozinha. De que não se precisa de nenhum conhecimento específico para operação própria, e de que, desde que se pague separadamente por cada operação, cada leigo dono do computador fácil de usar, possa escrever textos, fazer compras pela internet, e jogar jogos interativos.

1.2 Contexto Atual da Computação Pessoal

Na verdade o computador como caixa mágica só se tornou possível depois que suas capacidades estavam suficientemente desenvolvidas, em termos de memória, armazenamento, capacidade gráfica e outras, para que ele pudesse conter, sem que fosse necessário trocar disquetes, os “aplicativos de caixa preta”, como o processador de textos e o navegador para a Web.

Antes disso, em particular com os computadores pessoais que existiram ao longo da década de 1980, não era incomum aos usuários de computadores fazerem seus próprios programas. Em geral programas simples, que apenas permitiam explorar as capacidades do computador, mas as vezes coisas mais sofisticadas. Os próprios livros de informática publicados na época traziam programas curtos, para serem digitados pelos próprios leitores, tanto a título de exemplo como até programas gráficos ou jogos.

É interessante notar que ao contrário de hoje, haviam tipos diferentes de computadores que eram vendidos com linguagens de programação embutidas com pequenas diferenças entre si. Mas essas diferenças aumentavam quando se fazia acesso aos recursos gráficos ou sonoros de cada computador. Assim sendo, a maior parte desses livros era específica para um dos sistemas existentes. Havia revistas publicadas que pretendiam trazer alguns programas que funcionassem em todos os tipos de computadores. Para isso, dentro de um mesmo artigo, haviam sessões para cada tipo de computador popular na época, explicando as modificações necessárias ao programa para aquela máquina.

Hoje em dia, ao contrário de então, estamos reduzidos a um tipo de computador

mais amplamente utilizado nas casas e escritórios - o descendente do computador pessoal criado pela gigante da computação para empresas, a IBM. O "IBM PC" por questões de mercado acabou levando os outros a extinção, apesar de ser tecnicamente inferior em muitos aspectos. Sua adoção por grandes empresas nos escritórios lhe emprestou a fama de "computador sério" enquanto os outros eram vistos como brinquedos, aliado a uma estratégia de licenciamento para outras fabricantes acabou levando a sua universalização. Uma outra plataforma que sobrevive, mas que, sobretudo no Brasil, tem uso bastante restrito é o Macintosh da Apple.

As opções que se tem hoje dizem respeito ao Sistema Operacional que vai ser utilizado na máquina. Embora mais de 90% dos computadores utilizem o Windows da microsoft, a ponto de, para o público geral, o esse sistema ser confundido com o computador em si, ele pode ser substituído por outros sem qualquer alteração física no computador. O capítulo 2 se aprofundará nessa questão.

No entanto, os computadores de hoje não vêm com uma linguagem de programação embutida. As linguagens de programação diversas são implementadas por aplicativos comuns, que não são muito diferentes de outros aplicativos como os processadores de texto. Alguns sistemas operacionais vem com várias delas como parte integrante de si. Mas a filosofia de algumas dessas linguagens é justamente uma orientação para que as mesmas funcionem do mesmo modo em todos os tipos de computador, e em todos os sistemas operacionais. Assim, como os livros da década de 80 podiam ser divididos pelo tipo de computador a que se destinavam, os de hoje em dia podem ser pelo tipo de linguagem. No entanto, estes são voltados quase que exclusivamente para o público profissional da area de informática. Os programas recreativos dos livros e revistas da década de 80 se perderam para o público de hoje. E mesmo onde aparecem, programas simples, compreensíveis, e com algum efeito

prático, esses são mostrados como exemplos para que o profissional possa entender melhor a linguagem que se lhe apresenta, e não como fins em si, para que o usuário final o modifique a seu bel prazer.

Mesmo os projetos para uso de informática na escola que datam do final dos anos 80 refletiam isso³ - querendo trazer a programação de computadores para o currículo, já que esta era uma das únicas formas do computador ser interessante. Em particular uma programação que tivesse como resultados visíveis na tela quase diretos. Para tanto, vários projetos foram desenvolvidos utilizando-se a linguagem Logo - que permite uma proximidade com a geometria logo no início de seu aprendizado, e possibilitando que os alunos usassem a capacidade do computador de repetir centenas, milhares, de vezes os mesmos comandos para realizarem belos desenhos.

Os projetos não tiveram, em sua maioria, continuidade, e os computadores se tornaram máquinas que podem ser usadas sem que se tente ou que se queira, ou que se tenha a menor idéia de como, programa-los, e, conseqüentemente, programar virou palavirão nas ciências humanas e na educação.

Mas é justamente onde, além de servir como ferramenta para escrita e para a criação de artes, o computador vem trazer novidade - se arte, ou se apenas jogo, com certeza tem ao menos tanto valor quanto o jogo de xadrez apenas pelo seu valor lúdico.

³Por exemplo, o programa Eureka coordenado pela professora Afira V. Ripper (Ripper, 1996)

2 Liberdade e Corporatividade

2.1 A História da Incrível Máquina de Construir Casas

A título de introdução a esses conceitos, considere que houvesse máquinas que fossem capazes de construir casas. Essas máquinas seriam gratuitas, ou alugáveis um preço baixo, e poderiam construir ou reformar qualquer construção a um custo zero. Para tanto, precisariam que lhes fossem fornecidas apenas das plantas das casas a construir.

Nesse mundo hipotético, os arquitetos se dividiriam em duas classes - uma que para preservar seu status, trancaria todas as plantas de casas a 7 chaves, e faria questão de só vender casas prontas, construídas por eles próprios, e não forneceria a planta ao cliente. Outra que forneceria as plantas de bom grado, podendo ou não operar as máquinas para a construção final da casa. Com as plantas em mãos, o cliente poderia fazer as reformas que bem lhe aprouvessem, e implementá-las a custo zero. Poderia ainda repassar cópias das plantas para que amigos seus pudessem fazer suas casas, sem nada gastar.

Enquanto que tais máquinas seriam impossíveis no mundo físico, no mundo dos

computadores a situação acima é exatamente a que ocorre. No lugar de casas, se constroem programas de computador. E, dadas as plantas, essa construção se dá a custo zero. No entanto, até o presente momento, o costume é de se pagar pelo programa pronto, sem que se possa modifica-lo, sem que se possa sequer saber como ele opera.

2.2 Propriedade Intelectual

Neste capítulo, aborda-se um aspecto que a primeira vista pode parecer ser por demais técnico, ou sem qualquer importância para a educação. Mas como será fácil verificar, trata-se de assunto de vital importância para a formação do cidadão nos dias atuais, relacionado inclusive com aspectos não diretamente ligados à informática.

Trata-se sobretudo de propriedade do conhecimento.

Sem dúvida, uma expressão interessante - “propriedade do conhecimento”. Na verdade, as discussões atuais tanto na mídia impressa como nos devidos fóruns da internet, se centram em várias discussões sobre a expressão paralela “Propriedade Intelectual”. Nessas discussões, através de uma série de falácias - no sentido estrito do termo, isso é, uma conclusão aparentemente justificada pela lógica, mas que faz deliberadamente uso incorreto da mesma - faz-se parecer que a propriedade intelectual, defendida quase universalmente pela mídia, é algo estritamente bom, e deve ser aplicado sem restrições a todos os casos.

Na verdade a mídia dos noticiários, por estar intimamente associada a grandes

conglomerados corporativos que controlam outros tipos de mídia, e demais grupos que obtém vantagens financeiras e outras com a situação atual, chegam ao extremo de dizer que sem a devida proteção à propriedade intelectual, a música como forma de arte poderia acabar, e também, para se ter um panorama de quão ampla é essa questão, a capacidade humana de produzir medicamentos.

Como exemplo, pode-se ver essas duas citações, dentre várias facilmente localizáveis na internet.

“...Songwriters, singers, musicians, labels, publishers - the entire music food chain is at serious risk. The RIAA estimates that (...) 3.6 billion songs are illegally downloaded every month.

This problem won't be solved in short order. It's going to require education, leadership from Washington and true diligence to help our fans (...) to embrace this life and death issue and support our artistic community ...”

“Compositores, cantores, músicos, selos, editores - toda a cadeia alimentar da música está em sério risco. A RIAA estima que (...) 3.6 bilhões de canções são baixadas da internet ilegalmente todos os meses.

Este problema não será resolvido a curto prazo. Será necessário educação, liderança de Washington e verdadeira diligência para fazer com que nossos fãs abracem esta causa de vida e morte e suportem a nossa comunidade artística...”¹

¹GRAMMY Awards (2002) apud http://www.rattlehead.com/news/words/the_insidious_virus_of_illegal_music_downloading.html acessado em 2/12/2002)

“Alan Holmer, president of the Pharmaceutical Research and Manufacturers of America, criticized Clinton’s action: ‘We recognize that AIDS is a major problem, but weakening intellectual property rights is not the solution.’”

“Alan Holmer, presidente da Pharmaceutical Research and Manufacturers of America, criticou a ação de [do presidente] Clinton: ‘Nós reconhecemos que a AIDS é um grande problema, mas enfraquecer os direitos de propriedade intelectual não é a solução’ ”²

Ora, é fato incontestável que antes de existir a capacidade técnica de se gravar uma música, e reproduzi-la em massa, o que só ocorreu em fins do século XIX, a música já existia como forma de arte, e nunca se ouviu falar de um período da história da humanidade em que não existissem artistas ou músicos. O que não havia eram pop stars ganhando milhões de vezes mais que as outras pessoas apenas com sua imagem, e gravadoras e empresários ganhando mais algumas dezenas de vezes esse valor.

Quanto a indústria de medicamentos, seria um engano coloca-la no exatamente no mesmo barco. Tanto é que, se as músicas são cobertas por “direitos autorais”, essas se defendem com as “patentes- duas formas distintas de propriedade intelectual. De fato, as pesquisas pagas pelas indústrias privadas esperariam um retorno do investimento a partir das vendas daqueles medicamentos. No entanto, quando elas colocam esses valores como absolutos, ao ponto de perdas de vidas humanas serem um problema menor do que o retorno de um investimento já feito, nota-se que apesar

²<http://www.cnn.com/2000/WORLD/africa/05/11/aids.africa/> acessado em 2/12/2002 Sobre a produção de medicamentos genéricos para combater a doença novembro de 2000.

de não estarem no mesmo barco, estão certamente no mesmo rumo.

Na verdade, enquanto que no caso de músicas só se pode falar em propriedade intelectual, no segundo caso já se pode falar em propriedade do conhecimento, de forma mais restrita.

A quem pertence o conhecimento humano? Embora seja uma questão aparentemente vaga, uma pequena ponderação traz claramente a única resposta plausível: à humanidade.

O que ocorre, é que no modelo de sociedade atual, partes desse conhecimento são restritas à partes da humanidade. No caso de invenções e remédios não é tanto a sonegação de informações, mas sim, devido a patentes, que certos conhecimentos, apesar de publicamente registrados e catalogados, não podem ser colocados em uso sem a licença do detentor da patente.

Ao contrário do que possa parecer, e do que se quer fazer parecer pela indústria detentora de direitos autorais e outros tipos de propriedade intelectual, a formalização legal das leis de direitos autorais e patentes não se deu apenas para garantir aos inventores/criadores formas de obterem rendimentos através do monopólio da exploração comercial de suas obras, mas também, e desde o primeiro momento, garantir que a partir de um certo prazo considerado justo, essas obras passassem a ser de domínio público, vindo integrar o patrimônio cultural da humanidade.

De fato, num modelo de negócios livres, como querem os liberais, houve um momento histórico em que havia a necessidade de se recompensar materialmente os inventores de um dado método, ou uma descoberta científica com aplicação prática

na produção industrial, uma recompensa pelo trabalho depreendido com a pesquisa. Essa recompensa se dá na forma de patentes - o inventor tem o direito de utilizar os métodos que descobriu como bem entender, inclusive negociando o seu uso por terceiros, mediante o pagamento de royalties.

Dessa forma, havia sido criado um método razoavelmente simples de se mercantizar o conhecimento e a pesquisa científica. Ainda hoje, mais de um século depois da institucionalização das patentes, certas vertentes políticas se baseiam nas mesmas para tentar mover todo o tipo de pesquisa para fora da Academia (Universidade), e para dentro da indústria privada.

No entanto, o que fica óbvio nesse modelo de recuperação de investimentos em pesquisas através das patentes, mas que é ignorado pelos seus defensores, é que não se tem um modelo de investimento em pesquisa, mas sim, um modelo de jogo de azar em pesquisa. Um tipo de jogo de azar em que quanto maiores as apostas, maiores e quase certas - as chances de se ter um retorno financeiro. Mas os pequenos jogadores ficam de fora.

Por exemplo - se uma média empresa empreende uma quantidade considerável de recursos em uma dada linha de pesquisa, e dali não emerge nada imediatamente rentável para a indústria, seu investimento foi jogado fora. Por sua vez, uma multinacional que invista - em seus laboratórios centrais - em dezenas de linhas de pesquisas simultaneamente pode recuperar de apenas uma delas que resulte num produto lucrativo, os investimentos feitos em todas as outras.

2.3 Patentes de Software e Algumas Implicações

Se isso é perverso para o modelo industrial de produção, torna-se perversidíssimo ao se permitir patentes sobre idéias ou algoritmos, tais como as patentes de software. Isso por que esses algoritmos - que podem ser pensados como sequências de expressões matemáticas e lógicas - constituem o “esqueleto” dos programas de computador. As patentes permitem que alguém tenha o direito sobre o resultado final de um programa, sem diferenciar como ele implementa isso internamente. Portanto, se as patentes de software - hoje lugar comum nos EUA, e tentando se estabelecer na Europa - fossem válidas nos primórdios da computação, na década de 70, alguém poderia, por exemplo, ter patenteado o conceito de “uma tecla, ao ser premida no teclado, é codificada para uso interno do computador, e seu caractere correspondente é desenhado na tela”. Isso teria atrasado a computação interativa por 20 anos - uma vez que apenas os detentores de tal patente poderiam fazer computadores que operam como os que temos hoje. A xerox poderia ter patenteado a idéia do mouse para manipular objetos na tela, propiciando outro atraso de 20 anos. E assim por diante, de forma que possivelmente os computadores pessoais de hoje ainda usassem cartões perfurados.

Adicione-se a isso que o período de duração das patentes de software nos EUA e o mesmo que é usado para patentes “físicas”, desde o século XIX. Num ambiente em que a velocidade de processamento dobra a cada 18 meses, alguém se torna “dono” de um tipo de operação por 18 anos (na verdade 17). É completamente impossível se prever o estado da arte da computação para daqui a 10 anos. Menos ainda 17. Mesmo um dos maiores defensores de patentes de software, o dono da livraria virtual Amazon.com, Jeff Bezos, sugeriu em entrevista que esse tempo fosse

de cerca de 3 ou 4 anos.³

No entanto, este prazo, como para as patentes físicas, é de 17 anos nos EUA. No Brasil, não são permitidas patentes de algoritmos ou de software, e, o tempo de duração de uma patente é de 20 anos de acordo com a lei 9279/1996⁴, de patentes. Ou seja, uma nova invenção, método ou produto industrializável pode ser explorado com exclusividade por seu criador, ou melhor dizendo, pelo detentor da patente, por 20 anos. Só então outras pessoas ou empresas podem empregar a mesma técnica sem ter que obter autorização, normalmente sob a forma de royalties, do criador original da patente.

2.4 Direitos Autorais e seus Prazos

A questão de direitos autorais no entanto, relacionados a obras artísticas e outras, tais como livros, músicas e filmes, é diferente. Esses direitos, ao contrário dos 20 anos de uma invenção patenteada reverterem para domínio público 70 anos após a morte do autor original, no Brasil (lei 9.610/98). Nos EUA esse prazo é de 50 anos após a morte do autor, com uma discussão que se arrasta ao longo dos últimos anos para sua alteração para 70 anos. Alteração esta que acontecerá de qualquer forma se aquele país assinar ao tratado da Área de Livre Comércio das Américas - ALCA, na forma como se encontra o rascunho hoje⁵. Costuma se dizer que a cada vez que o personagem Mickey Mouse, pertencente à corporação de mídia Disney, está prestes a cair em domínio público, devido a expiração dos prazos de proteção contados a partir da morte de seu criador, o congresso daquele país altera a lei retroativamente,

³<http://www.bitsplace.com/bitsplace/BITSupdate31100.html> , acessado em 20/3/2003

⁴http://www.direitonaweb.adv.br/legislacao/lei9279_96.htm, acessado em 20/04/2003

⁵http://www.ftaa-alca.org/ftaadraft02/por/ngipp_1.asp#II.3.Artigo10, acessado em 30/05/2003

extendendo o prazo de proteção por mais vários anos.

No princípio, com a lei original de 1790, os autores dos estados EUA retinham por 14 anos os direitos sobre seus trabalhos, renováveis por mais 14 se estivessem vivos ao término do primeiro prazo. Em 1831 esse prazo foi aumentado para 28 anos, com possibilidade de renovação para mais 14. Em 1909, aumentado para 28 anos, renováveis por mais 28. Em 1976, foi alterado novamente para 56 anos após a morte do autor. Em 1992, a lei foi alterada para que a renovação dos direitos autorais fosse automática. Isso significou que muitos trabalhos não mais disponíveis há muitas décadas, e esquecidos pelos descendentes de seus autores, não puderam entrar em domínio público. E finalmente, em 1998, uma extensão adicional de 20 anos para os direitos autorais, fazendo com que nenhuma obra seja de domínio público até 70 anos depois da morte do autor.

É interessante enfatizar que menos de 2% dos livros impressos na década de 1920 estão em catálogo hoje, restando então a quem queira adquiri-los procurar por cópias cada vez mais escassas em lojas de usados e coleções particulares. Se essas obras já estivessem em domínio público, uma série de projetos iniciados na Internet para preservação da cultura, iria gradualmente criando versões digitais dessas obras de forma que elas fossem garantidamente preservadas para a posteridade, além de serem facilmente encontráveis tanto por seus títulos quanto pelos seus conteúdos e assuntos com que lidam, uma vez em formato eletrônico. No entanto, devido às últimas extensões desses termos de direitos autorais, nenhuma dessas obras pode ser disponibilizada eletronicamente por projetos como o Projeto Gutenberg⁶.

⁶<http://promo.net/pg/>

2.5 A Lei Além da Justiça: Proteção Contra Cópia

Além da questão de livros ou músicas, as grandes indústrias, ao desenvolverem novos suportes de mídia - para filmes ou outros - já o fazem pensando primeiramente em como impedir a cópia do conteúdo, para depois pensar em qualidade e praticidade do método escolhido. Ultimamente, de novo, primeiro nos EUA, com risco de se espalhar pelo mundo, tem-se um coquetel de impedimentos legais e tecnológicos, que visa tornar virtualmente impossível a cópia de qualquer produto “com direitos intelectuais” a ponto de ser muito difícil, talvez inalcançável, que se faça uma simples citação do trecho de uma obra, por exemplo.

Ao lançar o formato de vídeo em DVD, por exemplo, a indústria de conteúdo tentou fazer um formato digital com proteção contra cópia. Essa proteção é funciona mais ou menos da seguinte forma: os arquivos com os filmes em si permanecem criptografados no disco.

Há algumas chaves secretas que podem descriptografar os arquivos, e tornar o conteúdo acessível. Essas chaves eram segredo da indústria de conteúdo, que, mediante pagamento e assinatura de contratos, podiam fornecer uma dessas chaves a uma empresa que se dispusesse a colocar no mercado um “tocador de DVD”. Quer só o software, na forma de um programa binário, quer um “eletro eletrônico- o aparelho DVD Player que fica embaixo da TV.

Note que por essa concepção é difícil imaginar um programa de software livre - do qual sempre se tem “a planta- que pudesse “tocar” um DVD. Se a chave secreta estivesse no código fonte do programa, poderia ser lida por todos. Ou ainda, o código fonte do programa pode ser alterado de forma que em vez da saída do filme

descomprimido se dar para a tela, que essa saída seja feita para um arquivo contendo o vídeo limpo, legível por qualquer outro programa e indefinidamente reprodutível.

Há formas de se fazer esse “controle de direitos autorais-(DRM - Digital Rights Management) , através de software livre, através de usos um pouco só mais sofisticados das técnicas de criptografia. De fato, no dia 4 de abril de 2003 mesmo surgiu anúncio de um grande projeto para se adicionar DRM com software livre ao formato Ogg Vorbis, que implementa a música digital compactada, como a dos populares arquivos MP3, mas sem estar restrito por patentes de software⁷

Mas no caso do DVD se o consumidor compra um disco de DVD, ocorre o seguinte:

Dentro da lógica industrial, de coisas manufaturadas, a pessoa comprou é dele, e ele pode fazer o que quiser. Na lógica distorcida do software proprietário, onde se tem pegar o pior dos dois mundos -

1) O pior da lógica industrial para bens manufaturados: ele comprou um, tem um. Se for copiável de graça , ele só pode continuar tendo um, por que só pagou por um.

2) O pior da lógica do modelo de serviços: Ele não comprou de fato. O direito de assistir ao filme é cedido a ele mediante o pagamento do preço do DVD. Ele pagou pela licença de assistir ao vídeo, não de reproduzi-lo (num sentido diferente do de assistir).

Comparando-se como isso funcionaria numa lógica que se adequaria ao mundo

⁷<http://slashdot.org/article.pl?sid=03/04/03/2059223>, acessado em 15/04/2003

dos produtos digitais:

A) ele pagou para ter acesso ao conteúdo, e pode fazer o que quiser. Se é infinitamente copiável, ele pode fazê-lo.

Jon Lech Johansen, um Norueguês, na época com 16 anos, teve acesso à chave criptográfica para leitura de DVDs. Criou um programa - o DeCSS, o que permitia leitura e decodificação dos mesmos, e o distribuiu na Internet na forma de código fonte (planta do programa).

Mas o fato é que não usando software proprietário, "DVD Jon", como Johansen se tornou conhecido na mídia, não podia nem usufruir do direito de assistir à cópia pela qual havia pago. A conferir: ele pagou pelo direito de assistir ao DVD, pagou pelo dispositivo físico - o leitor de DVD - mas não podia assisti-lo por não haver um decriptografador de DVD para o seu sistema operacional.

Então, Johansen conseguiu escrever um tal decriptografador. Desta forma ele pode assistir seus DVD's, ainda dentro do que poderia fazer sob a restrição "2" acima.

O que os Norte Americanos fizeram em seu próprio país, e tentam agora impor ao mundo, é o DMCA. "Digital Millennium Copyright Act".

Entre as demais restrições impostas ao DMCA, há o equivalente legal de:

3) Se em algum produto de mídia houver um dispositivo tecnológico que impeça que este conteúdo seja copiável, o ato de contornar esse dispositivo permitindo a cópia do conteúdo é ilegal, bem como o ato de prover formas de se contornar esse

dispositivo legal⁸

Além do DMCA, de 1998, já andou circulando pelo congresso dos EUA uma CBDPTA - "Consumer Broadband and Digital Television Promotion Act"⁹.

uma lei que acrescentaria:

4) Todos os fabricantes de produtos digitais são obrigados a embutir em seus produtos tecnologias para impedir que produtos de mídia cuja cópia seja protegida por direitos autorais sejam copiáveis.

Isso incluiria discos rígidos, monitores, tudo. De fato, a programação tal como nós a conhecemos não seria mais possível, já que pelo menos no projeto que circulava em 2002, esse impedimento de cópia abrangeria inclusive o que já existisse anteriormente em formatos analógicos, ou digitais, mas sem as proteções criptográficas como as do DVD. Ou seja uma simples cópia de um arquivo de texto de uma pasta para outra no próprio computador pessoal seria um comando impossível para a maioria dos usuários, já que o sistema não teria como saber se o arquivo texto em questão não seria um texto protegido por direitos autorais.

Derivado da idéia do CBDPTA - uma sigla que aliás, parece ter sido propositalmente pensada de forma a não poder ser lembrada com facilidade, justamente para complicar as discussões a respeito - aparece o TCPA - Trusted Computing Platform Architecture - uma iniciativa suportada por cerca de 150 empresas, e sua implementação, da Microsoft, conhecida antes como "Palladium", e renomeada para algo como "Microsoft Next-Generation Secure Computing Base- em que as limitações do

⁸Minuta da ALCA, capítulo Sobre Direitos de Propriedade Intelectual, artigos 20 e 21 http://www.alca-ftaa.org/ftaadraft02/por/ngipp_1.asp#DPI, em 13/05/2003

⁹<http://www.everything2.net/index.pl?node=CBDTPA> - em 20/03/2003

(4) se fariam presentes de forma aparentemente mais amena -é possível se usar o computador para copiar arquivos ou dados não protegidos por essa tecnologia. Mas se um arquivo de mídia estiver num formato protegido de acordo com a TCPA, torna-se impossível, devido a toda uma hierarquia do sistema construída desde o hardware e através de várias camadas de software, que o mesmo seja copiado, ou modificado através de um programa - a única coisa possível é que ele seja visto numa tela, ou ouvido pelos auto-falantes. E aproveitando o ensejo, a tecnologia permitirá mesmo que se inclua um prazo para que o arquivo seja visto ou ouvido, ou um número máximo de vezes para sua exibição.

Todos esses “avanços tecnológicos”tema intenção de serem implementados de forma invisível e onipresente em todos os sistemas computacionais pessoais. Uma vez que a empresa Microsoft goza de monopólio do sistema operacional para os computadores pessoais, e uma vez que a mesma não tem por costume informar esses planos, quer no estágio atual de desenvolvimento, quer na época em que os mesmos estarão embutidos na nova versão de seu sistema “Windows”, efetivamente milhões de pessoas estarão “voluntariamente”se trancando atrás de grades digitais cuja própria existência seria impensável para as pessoas do meio de alguns anos atrás.

Uma vez que os sistemas digitais são hoje o que é usado para a escrita formal efetiva, como se mencionou no capítulo anterior, as implicações de tais medidas “anti pirataria”representam de fato um perigo iminente para toda a cultura da humanidade. Ora, não se conhece hoje quem, dos que participam do mundo da escrita formal, não tenha se defrontado com a “correção automática”do Microsoft Word alterando automaticamente a grafia de uma palavra, ou transformando um trecho de texto numa “lista de tópicos”. Num futuro próximo, as exatas mesmas carac-

terísticas do programa podem ser usadas, por exemplo, para impedir que algumas palavras sejam escritas - imagine-se uma "lei anti-palavrões- com as opções de desabilitar o controle dos software sobre o texto digitado removidas por uma combinação de forças legais e tecnológicas.

2.6 Formalizando o Software Livre

Transpondo esses conceitos para a realidade mais genérica dos programas de computador, temos justamente a propriedade intelectual como um dos eixos que pautam as relações de produção e de comércio, ou distribuição, dos mesmos. Essas idéias de propriedade intelectual na informática são reforçadas pelo fato de que programas de computador, bem como outros trabalhos existentes em forma digital, não dependem de seu suporte físico, sendo indefinidamente reproduzíveis. Isso é: se você tem como executar um programa de computador, também tem como copia-lo. A operação de cópia pode ser dificultada por artifícios técnicos, mas sempre será possível pela própria natureza do funcionamento dos computadores digitais.

Portanto, ao se tentar simplesmente transpor um modelo de economia industrial existente para o universo da informática, fez-se uso da propriedade intelectual para emular a propriedade física de um produto. Se uma pessoa vai ao supermercado e compra uma bola, ela é dona dessa bola. Se quiser vender bolas iguais aquela, terá que fabrica-las, deprendendo tanto trabalho no processo quanto o fabricante original. Daí não haver, exceto no caso de patentes, preocupação com a propriedade intelectual de objetos físicos. No caso de um programa de computador, no entanto, o comprador tem os meios para efetuar quantas cópias desejar do programa, e passá-las adiante. Não havendo limitação física para esse ato, cria-se uma limitação legal,

aplicando-se aos mesmos as leis de direitos autorais. Assim, a cada cópia de um programa em uso, deve corresponder, nesse paradigma, uma cópia vendida pelo seu criador original.

No entanto, a natureza dos programas de computador difere por demais da dos produtos manufaturados, para que uma réplica do paradigma de “pague um leve um” funcione corretamente com os mesmos. De fato, enquanto que esse seja o modelo predominante para os programas em uso hoje, o que ocorre é que mais da metade dos programas instalados no Brasil são na verdade cópias ilegais - isso é, que estão instaladas mas não correspondem a uma cópia paga ao fabricante. A razão, como já foi dito, é que é empregado trabalho humano para primeiro se fazer um programa. Depois que ele esteja funcional, sua replicação é, em termos práticos, gratuita.

Um programa de computador, em sua forma final, é uma sequência de dados numéricos que pode ser diretamente interpretada pelo computador, que interage com outros programas, que compõem o Sistema Operacional. Ele provê em geral alguma funcionalidade a mais para o computador, conforme será aprofundado no capítulo 4. No entanto, como um arquiteto que faz a planta de uma construção, não a construção em si, os programadores de computador não escrevem o programa na sua forma final, como ele é interpretado diretamente pelo computador.

Os programas são primeiramente feitos em alguma linguagem, computacional, que é sempre legível por seres humanos. Então esses programas, nesse estágio chamados de programas fonte, são processados por um tipo de aplicativo, chamado compilador - este aplicativo, na verdade, um outro programa, traduz o programa fonte, gerando um outro arquivo - o programa objeto, também chamado de exe-

cutável, que faz sentido apenas para o computador alvo, não sendo mais inteligível.

A partir desse ponto, o programa objeto pode ser copiado e instalado em qualquer número de computadores, mas em geral, não pode mais ser modificado. Se se deseja fazer alguma alteração, apenas a pessoa que possui o código fonte - como foi dito, o equivalente à planta de uma casa - pode fazer a alteração, e gerar, de forma praticamente instantânea, um novo arquivo executável.

A partir daí, a indústria tradicional de software, baseada no modelo da indústria de bens de consumo, por inércia oferece apenas os arquivos executáveis quando “vendem” os seus produtos. Se tomarmos por analogia um carro, ao comprarmos um, também não temos acesso aos projetos. Mas com um carro, não basta alterar alguma especificação no projeto, apertar um botão, e esperar o carro ser substituído por uma versão corrigida.

No entanto, com programas de computadores, o caso é justamente esse. Pode se fazer uma alteração no projeto, e se usufruir da mesma, sem nenhum custo adicional.

Deve-se ter em mente que se os carros em geral não são acompanhados de seus projetos, o mesmo não se pode dizer sobre casas. Nesse caso, os compradores tem acesso ao projeto, e podem fazer as alterações que bem desejarem, respeitadas as devidas limitações.

Assim, quando surgiram os primeiros computadores comercializados - antes dos computadores pessoais, quando esses eram restritos a grandes corporações ou instituições acadêmicas - que ocupavam a maior parte de uma sala, os programas para operação dos mesmos eram tratados de forma secundária. Saber fazer um programa

para operar o computador era o equivalente a mudar a estação e o volume de um receptor de rádio: um conhecimento que era naturalmente dividido entre todos os envolvidos.

Foi somente com o passar do tempo, e com números maiores de computadores em operação, que surgiu a idéia de se vender o software separadamente. E depois disso ainda surgiu a idéia de não mais se fornecer ao cliente o código fonte dos programas. Mas de forma paralela, no meio acadêmico, partilhar o software sempre foi a norma.

Então, por exemplo, se alguém escrevesse um programa para copiar um arquivo, e outra pessoa precisasse de um programa para mudar todas as letras minúsculas de um arquivo para maiúsculas, essa segunda pessoa não precisaria escrever seu programa a partir do nada. Ela poderia tomar o projeto do programa de cópia, e nele, introduzir apenas uma linha a mais de código para produzir a transformação desejada no arquivo copiado. Refazer todo o programa seria reinventar a roda.

Enquanto que um programa como o desse exemplo é extremamente simples, seus princípios permanecem válidos para qualquer complexidade de programa. À indústria de software atual, parece natural que cada novo projeto que se queira iniciar deva ser iniciado do zero, refazendo-se o trabalho de, por vezes, milhões de linhas de código - já que essas empresas não permitem que se faça uso, nem que sequer se tenha acesso, ao código fonte de seus produtos.

Esses programas são vendidos como se fossem um bem material comum - tijolos, iogurte, ou outro. Para um produto com existência física, se tem-se o desejo de fazer uma réplica do mesmo, é necessário empenhar tanto, ou mais trabalho, quanto o

vendedor original. É inconcebível que um fabricante de tijolos venha a processar alguém por estar fazendo réplicas de seus tijolos quer para uso próprio, quer para comercialização - desde que seja respeitada a marca. Com software, acontece que os "bens" são infinitamente reproduzíveis a custo zero. Todo o trabalho empregado se dá durante a criação do mesmo. Ao revende-lo por um custo alto para centenas, ou milhares de usuários, o investimento se paga as mesmas milhares de vezes. O custo para gravação de um CD e da embalagem é negligível se comparado as dezenas, as vezes centenas, de dólares que se paga por um programa na loja.

Não deixa de ser um modelo funcional, e inclusive justo, sob a forma de pensar estabelecida pela sociedade industrial. Mas, vejamos, se numa construção tem se que empregar 500 tijolos, tem que comprar os 500. Se você faz um tijolo, pode usar o seu, e comprar apenas 499. Por outro lado, se você tem um parque de 20 micro-computadores instalados numa rede, precisa comprar 20 processadores de texto. Se você faz um processador de textos - não precisa comprar outros 19. O seu programa você pode copiar e usufruir sem qualquer taxa extra nesses 20 computadores, como pode doa-lo a quem dele quiser fazer uso.

Mas quem vai efetuar a doação de um software não precisa se limitar a doar o programa executável. O código fonte, cujas cópias são tão gratuitas de se obterem quanto o outro, pode oferecer aos usuários finais uma série de de benefícios.

É interessante notar que quando esses conceitos são apresentados assim, tem-se a impressão de que bastaria que houvesse um programa de cada tipo disponível gratuitamente para que ninguém mais comprasse programas.

Não é isso o que ocorre. De fato existe uma versão em software gratuito, e livre,

de quase todos os aplicativos mais usados. No entanto por questões de desconhecimento, marketing, comodismo, entre outras, esses aplicativos são relativamente pouco conhecidos.

Ora, qual seria então a relação entre esses modelos de produção de software e a educação? De fato, essa relação não surge naturalmente até que se toque no assunto de transmissão do conhecimento. Entretanto, a educação escolar de hoje, se pauta justamente nesse fundamento, o de transmissão do conhecimento. De fato, temos hoje uma finalidade prática para a educação dos níveis fundamental e médio que é processo seletivo para o ensino superior. Em teoria entretanto, o fim da educação seria a formação do ser humano como um todo, algo não desconexo - sempre na teoria - da *paidéia* grega, em que o conhecimento das várias disciplinas escolares compõe uma parte da educação completa. Senão por outros motivos, a simples ausência de qualquer ensino filosófico, a segregação da “educação física” como uma disciplina a parte, e a única que trabalha o corpo, fazem do ensino contemporâneo no Brasil, e em muitos outros países, algo mecânico.

Mas é justamente nesse aspecto de transmissão do conhecimento - quase reduzido à transmissão de informação - é que surgem os paralelos necessários entre os modelos de produção de software e o ensino. Pois se é válido dizer que a humanidade apenas subsiste enquanto civilizada devido a uma manutenção e desenvolvimento gradual de seu corpo de saberes, transmitidos em maior ou menor grau a todo ser humano por meio do ensino, é válido dizer que se os saberes inerentes aos programas de computador que todos utilizam não forem compartilhados, os aspectos dependentes dos mesmos na humanidade, entrarão em colapso.

Em particular, ao se sustentar o paradigma do código fonte fechado, e que cada

novo programa deva ser escrito do zero, faz-se o equivalente a se sustentar que a cada nova máquina que se vá construir, tenha-se que - literalmente - re-inventar a roda.

Em contraposição a esse modelo disfuncional de produção e distribuição de programas, existe a idéia do programa de computador como serviço , não como produto. Isso é, paga-se pelo trabalho executado, a cada vez que é executado. Estando o programa pronto, ele passa a ser distribuível como o que é: um conjunto de informações indefinidamente copiável. Ora, se uma indústria tem condições de impor barreiras legais de forma que possa ser paga a cada vez que o programa é executado, e não uma única vez pela confecção original do programa, é natural - na forma como se está acostumado a pensar - que ela o faça. Se for dito, no entanto, que um pedreiro deveria continuar a receber dinheiro por uma parede levantada a cada vez que alguém entra no cômodo construído, já se acharia que é um ato estranho, talvez demonstrando excesso de ganância por parte do pedreiro, e e absolutamente injustas as leis que permitem tais pagamentos.

Mas além da questão do preço, o software livre dá ao seu usuário controle sobre o seu equipamento. Efetivamente alguém utilizando software livre mantém o direito de dizer ao computador o que fazer, sem entregar esse direito a uma empresa terceira. Se a este usuário faltar a capacidade técnica de exercer este controle, o fato de ter o código fonte a disposição permite que ele se associe com outros usuários ou, se for o caso, contrate os serviços de um terceiro - programador - para realizar as alterações desejadas. Com as taxas de licenciamento praticadas atualmente, tal serviço por parte de terceiros não excederia 3 ou 4 vezes o preço de uma licença.

Assim, retornando a um dos exemplos mais nefastos dados anteriormente, no

caso de uma “lei anti-palavrões”, estariam sempre garantido aos usuários os meios para escrever os tais palavrões, mesmo que fosse apenas em documentos “fora da lei”. E se uma lei que proíba a escrita de palavrões pode parecer pouco provável, tendo em vista o espaço que algumas corporações, e associações de corporações tem alcançado junto ao meio público, uma lei sutil que proíba que se escrevam informações sensíveis ou esclarecedoras sobre as mesmas, o que a imprensa de massa infelizmente já parece ter adotado para si, talvez não seja tão improvável.

3 Escrevendo e Publicando

3.1 A Escrita no Computador

Como já foi dito no primeiro capítulo, a maior transformação da informática na vida culta moderna se dá justamente na modificação dos hábitos de escrita, que ocorreu ao longo dos últimos dez anos.

A ferramenta que se utiliza para escrita é comumente um processador de textos, que inclui vantagens óbvias sobre a máquina de escrever, e a vantagem de fornecer um texto imprimível de excelente qualidade, no corpo e tipo de letra da preferência do autor, sem trabalho adicional.

Querer sumarizar aqui de forma objetiva todas as modificações trazidas ao hábito da escrita a partir do uso de um processador de textos seria fugir do escopo deste trabalho. Os leitores certamente reconhecerão, cada um, as modificações que se deram em sua própria maneira de escrever.

É certo que um grande número de pessoas, possivelmente a maioria, escreve em papel seus textos, de forma mais ou menos elaborada, antes de efetuar a digitação dos mesmos. Este é um hábito que deve se modificar, não só a medida que as novas

gerações de pessoas - e se não digo escritores é por que todas as pessoas escrevem, em menor ou menor grau, nem que só durante um período de aprendizado de sua vida - vão tendo que usar o computador para apresentar as versões finais de seus escritos - diferente de manuscritos -, mas também a medida que as interfaces para escrita forem se modificando, e se aproximando do que seja mais próximo da melhor maneira de se escrever.

Já hoje existem aparelhos, os chamados PDA (Personal Digital Assistants) que utilizam uma caneta especial como dispositivo de entrada, e na prática funcionam como um bloquinho de notas. Enquanto que na época atual, a forma de escrita utilizada é estilizada, tendo em vista que os softwares existentes são limitados para reconhecimento da escrita cursiva, esta é uma limitação que deve desaparecer ao longo dos próximos anos. Ironicamente, o primeiro modelo de PDA lançado, o Newton da APPLE Computers em 1993, fazia reconhecimento da escrita cursiva, sem apelo a modificação da forma das letras e à escrita de uma letra por vez, como os fabricantes de hoje implementaram a tecnologia. O motivo dessas modificações para os aparelhos da geração atual é que o reconhecimento de escrita do Newton deixava um pouco a desejar.

Uma outra grande modificação aos hábitos de escrita surge por parte do recurso de correção ortográfica automática, existente nos processadores de texto atuais. Enquanto que facilitam a entrega de textos sem erros ortográficos, é dispensável a realização de uma verificação objetiva para perceber que esse recurso causa um relaxamento nos hábitos de escrita, dispensando o escritor de se preocupar com a grafia correta das palavras que tenha dúvida. Esse relaxamento é acentuado quando se ativam os recursos de correção em tempo real, em que as palavras grafadas incorretamente são imediatamente destacadas no texto a medida em que são digitadas.

Caso haja dúvidas, convido o leitor a trabalhar durante as próximas semanas com esse recurso desligado, caso faça uso normal dele, ou com ele ligado, caso não o utilize.

Já foi dito que essa transformação foi radical - na grande maioria das situações de escrita formal, deve-se entregar um texto digitado eletronicamente no computador, quer impresso em papel, e, de forma crescente, no próprio arquivo digital, sem suporte físico, através da internet.

Isso leva também a algumas das questões discutidas no capítulo 2: se o texto é entregue como um arquivo digital, ele deve estar num formato padrão de textos. Se estiver num formato específico gerado por um editor de textos, principalmente um cuja empresa criadora decida não divulgar o modo como o conteúdo do texto é armazenado em seu arquivo, a pessoa que for receber o texto no formato digital é obrigada a ter um processador de textos da mesma marca, para poder abrir o arquivo.

Esta situação seria equivalente a que fosse necessário para ler um texto escrito com uma caneta Mont Blanc, o uso de uma outra caneta Mont Blanc. E de fato, por falta de informações, é o que acaba ocorrendo, uma vez que o processador de textos mais utilizado atualmente incorre nesses problemas. Tanto quem escreve quanto quem lê deve ter uma licença de uso do software, paga a uma única empresa. Os usuários sequer se dão conta de que tem apenas uma licença de uso do software - tem o costume de pensar no software como sendo seu - como se pode ter uma cadeira, ou um vaso de flores. Não são seus direitos legais, e as Licenças para o Usuário Final dos softwares proprietários se tornam mais restritivas ano após ano¹.

¹(<http://www.around.com/agree.html>; acessado em 17/11/2003)

Quanto a educação, parece claro que se ao longo dos últimos séculos um dos primeiros objetivos da educação formal é justamente a alfabetização - aí inclusos a leitura e a escrita - com uma transformação tão grande do ato de se escrever formalmente, em algum ponto no currículo essa escrita junto ao computador deve ser inserida.

Não se propõe aqui, de forma alguma, que o primeiro contato com a escrita deixe de se dar através de um meio físico, escrevendo-se diretamente sobre papel. Nem que a escola seja essencial para que se aprenda a digitar - se é que a interface permanecerá uma que requeira digitação - no entanto, parece apropriado, ao menos no ponto atual, que a escola fale sobre a representação do objeto escrito dentro do computador - o arquivo - como ele é copiado e movido, independente do sistema operacional, ou interface final, de forma que uma pessoa que escreva ao computador tenha conhecimento de onde se encontra o texto que escreveu, e o que tem que fazer se quiser envia-lo para alguém, imprimi-lo, ou apaga-lo sem deixar traços, conservando para si uma cópia num sistema de armazenamento portátil (ex. diskette).

Ainda que, como mencionado acima, a disponibilidade do corretor ortográfico faça com que a memorização da grafia correta das palavras seja enfraquecida, isso ainda será em muito menor escala do que foi enfraquecida a memória humana como um todo com a introdução da escrita, já que os gregos mesmos até Sócrates memorizavam todos os textos clássicos que hoje temos escritos.

3.2 A Escrita Eletrônica como Produção Pessoal do Aluno

Por outro lado, há a possibilidade de se escrever mais e melhor. Com a facilidade de se ter um público leitor, que através da internet, quer através da impressão e fotocópia dos textos, aumentam os estímulos à produção espontânea de material escrito pelos alunos. Se na educação em que os processos de escrita são puramente manuais essa produção se deve quase que exclusivamente aos textos obrigatórios exigidos pela disciplina de língua portuguesa e literatura, a criança e o adolescente que crescem hoje em contato com o computador certamente fazem um uso maior da escrita, mesmo que seja apenas para contato direto com seus amigos através de e-mail, ou de programas de mensagens instantaneas.

Estes últimos - programas de mensagens instantaneas - compreendem toda uma classe de programas na qual uma frase curta é escrita, e, se o receptor se encontrar com o respectivo programa em funcionamento, recebida instantaneamente. Um fato curioso é que isso mescla algumas características da comunicação oral à linguagem escrita, como imediatismo de resposta, abreviações, e mesmo vícios de linguagem. O mesmo se dá nas tão populares “salas de bate papo- ambientes virtuais, frequentemente embutidos em uma página de web, em que várias pessoas conectadas ao mesmo tempo trocam mensagens entre si, se representando por um nick - um apelido - virtual.

Não há dúvidas de que cabe ao educador perceber essas novas modalidades de escrita que passam a pervadir o cotidiano dos alunos, e integra-las ao currículo escolar, de forma a manter o interesse dos alunos pelo desenvolvimento da escrita, e seu uso de forma frutífera. Os usos acima tanto podem ser para uma comunicação

frívola, que do ponto de vista do educador e da sociedade em geral pouco teria a acrescentar a formação do indivíduo, como, com uma simples mudança de tema da sala de chat, por exemplo, para uma construção interativa, coletiva e democrática de temas bastante instrutivos para o indivíduo, não só em termos do conteúdo específico, que pode variar de botânica a astronomia, de jogos lógicos a música erudita, e assim por diante, como também em termos de uma atividade onde se aprender a ter respeito pela opinião alheia e a construção coletiva do conhecimento.

No entanto foi dito que essas formas de escrita que transmitem mensagens instantaneas são bastante contaminadas pela linguagem oral. E um dos fatores de contaminação é sem dúvida a efemeridade de uma discussão do gênero. Ainda que alguns minutos específicos numa sala de bate papo temática com 15 participantes não configurem uma situação excepcional, é difícil acompanhar todas as mensagens escritas com a atenção devida. E embora seja possível, no mais das vezes, se gravar a sessão para uma leitura posterior - um log- isso raramente é feito.

Uma sugestão possível a um educador lidando com uma situação dessas é tentar mover o fórum de discussões de algo tão evanescente quanto trocas de mensagens em tempo real para uma discussão por e-mail, ou um fórum na Web. Essas outras modalidades de escrita, sobretudo por perderem parte do imediatismo - o tempo para se elaborar uma resposta passa a ser de vários minutos até vários dias, em comparação com alguns segundos, numa modalidade de mensagens instantâneas, - passam a ser mais profundas, e principalmente, ficam registradas de forma recuperável para revisão, ou para serem lidas por outras pessoas que venham a se juntar ao debate.

3.3 A WEB

A World Wide Web, ou Web, ou mesmo a rede, como costumamos apelida-la, tem atualmente um impacto maior nos hábitos de leitura do que nos de escrita.

Primeiro é necessário esclarecer que a World Wide Web e a Internet não são a mesma coisa, como se costuma pensar. A Internet é a rede mundial de computadores - máquinas de todos os tipos interligadas por um protocolo de rede. Através da mesma, os computadores podem acessar uns aos outros, isso é, qualquer computador conectado à internet pode acessar qualquer outro (com algumas exceções). E isso é a internet. O que os computadores fazem quando acessam outro podem ser várias coisas diferentes, como transferir uma página da Web entre si, ou um e-mail, ou um fluxo de vídeo, ou iniciar uma sessão de bate papo entre os usuários em cada extremidade. Todas essas coisas são serviços de rede, serviços que funcionam sobre a internet, mas não são a internet.

Essa rede conectada, essa Internet tem uma história que remonta à década de 60, entre os militares e acadêmicos Norte Americanos². Na época, os serviços existentes basicamente se restringiam ao e-mail. Com o tempo foram surgindo outros serviços, como um tipo de boletim de notícias chamado Gopher, os chamados Newsgroups - espaços onde as mensagens são postadas como e-mail -, mas são de acesso público a todos os interessados, bate papo - o IRC -, e um dos serviços mais utilizados antes do advento da Web, o FTP - File Transfer Protocol ou Protocolo para Transferência de Arquivos - este protocolo permitia que arquivos fossem trocados, fazendo-se o uso de servidores, entre os vários computadores ligados na internet. Como hoje pode-se baixar uma arquivo de uma página da Web, com o FTP pode-se baixar um

²<http://www.zakon.org/robert/internet/timeline/> acessado em 17/11/2002

arquivo de um repositório de arquivo, sendo a maior diferença que não havia texto circundando os arquivos nesse repositório - só os nomes e os tamanhos dos arquivos, um embaixo do outro. Com a Web, pode se ter o mesmo tipo de serviço, mas com uma pequena sinopse de cada arquivo, por exemplo.

Na verdade hoje em dia se está acostumado a acessar a Web através de uma página que é um “portal” para as outras que se pretende utilizar - uma espécie de índice onde tudo são links. Ao se chegar ao texto de destino, uma notícia, ou uma portaria governamental, o que existe é texto corrido, sem mais links embutidos no texto - embora esses existam em barras laterais e no topo das páginas, onde ficam vários links que são utilizados para navegação dentro do site, quase nunca relacionados ao texto.

Mas a Web é uma construção peculiar. Seu crescimento se dá impulsionado por forças diversas, que a empurram em várias direções simultaneamente. Não só o crescimento bruto, por assim dizer, em termos de conteúdo e de computadores conectados, mas o crescimento técnico - o que é possível de ser representado numa página de web. De um lado os idealizadores e acadêmicos originais, de outro, o público que com sua popularização se tornou usuário e sobretudo leitor, e por fim, as empresas que tentam divisar várias maneiras de torna-la um negócio rentável, por si só, ou como suporte para aquisição de outros serviços, à maneira de uma loja.

Assim, quando foi proposta originalmente em 1990 no CERN (Conseil European pour la Recherché Nucleaire - Conselho Europeu para Pesquisa Nuclear) em Gênova na Suíça, seu propósito era unificar as informações na própria instituição, que com milhares de cientistas trabalhando concorrentemente, com estadia média de dois anos, se tornava caótica a ponto de recém chegados levarem muito mais

tempo tentando se interar do que estava acontecendo do que trabalhando em suas pesquisas³. Tim Berners-Lee escreveu sua proposta para o sistema que organizaria essas informações em 1989 como algo que pudesse ser “como um diagrama com círculos e flechas, onde os círculos podem ser qualquer coisa (...) podemos chamar os círculos de nós e as flechas de links” (idem). Nessa época o sistema ainda não estava batizado e Berners-Lee se referia a ele apenas como “Mesh” (estrutura). Foi apenas ao escrever o código de programa para sua implementação, no ano seguinte, que Berners-Lee o batizou de “World Wide Web”.

No início o padrão HTML (Hiper Text Markup Language) desenvolvido por Berners-Lee para a World Wide Web, permitia apenas a formatação do texto básica do texto com títulos, sessões, listas, e principalmente links. Ou seja, não estava presente suporte a cores, imagens, posicionamento do texto, tabelas, e muito menos animações e sons.

O funcionamento da WEB é a chave para entendermos como as pressões para sua modificação foram ocorrendo. Ela funciona basicamente com dois programas: um servidor, e um cliente de HTML. O servidor é o programa que fica “na internet” e cliente é o programa que fica no computador do usuário - do “internauta”.

Quando se acessa um site, o usuário digita o endereço que da página que deseja visitar no seu programa cliente, que é normalmente chamado de navegador. Hoje em dia, o mais comum é o Internet Explorer (proprietário). Esse programa então contata pela internet o endereço destino, e manda dados num protocolo chamado HTTP (Hyper Text Transfer Protocol) requisitando a página desejada. No computador localizado no endereço de destino na internet, está sendo executado o programa

³Berners-Lee, Tim, <http://www.w3.org/History/1989/proposal.html>; acessado em 15/10/2002

servidor. Este recebe os dados no enviados pelo computador do usuário, e, se forem a requisição para ver uma página, por exemplo, ele acessa esses dados armazenados localmente, isso é no mesmo computador em que está rodando o servidor, e os transmite para o computador que está executando o programa cliente. Esses dados enviados pelo servidor estão na linguagem HTML.

O programa cliente então interpreta a linguagem HTML para formatar o texto contido no documento e exibi-lo na tela, ou imprimi-lo.

A parte interessante é que nem o programa servidor nem o cliente são programas específicos. Podem ser vários programas, escritos por empresas, ou pessoas diferentes. O que está definido são os padrões: HTTP, HTML. e URL Qualquer programa escrito para ser um servidor WEB que respeite os padrões especificados para o HTTP funcionará bem como tal. Da mesma forma, qualquer programa cliente que possa se comunicar pelo protocolo HTTP, e interpretar a linguagem HTML é um bom programa cliente. O padrão URL - Unified Resource Location - define o modo de endereçamento de qualquer recurso na Internet, inclusive o de páginas Web. É basicamente um endereço padronizado em que entram o serviço da Internet sendo usado, o nome do computador, e sua localização na rede (domínio), e o endereço do arquivo, ou recurso, dentro daquele computador.

Então, não muito tempo após a colocação em uso no CERN por Berners-Lee de seu primeiro servidor, começaram a surgir outros programas, que eram compatíveis com essas especificações.

Em particular, Pei Wei, na universidade de Berkeley, adaptou um projeto pessoal que já era capaz de mostrar imagens e textos para ser também capaz de interpretar

a linguagem HTML. Este programa tornou-se o ViolaWWW ⁴, o primeiro Browser a aceitar textos e gráficos em conjunto. Mais tarde, Marc Andreessen ⁵, escreveu um programa chamado Mosaic. Este aceitava, além do HTML especificado por Berners-Lee, a inclusão de imagens nos documentos, como o Viola, mas também arquivos de som e outros tipos de documento.

Mas como esses browsers podiam expandir a linguagem? Por que o HTML é essencialmente uma linguagem transmitida como texto. Os comandos específicos para formatação ficam entre os sinais `<` (menor que) e `>` (maior que), como `<H1>`, `</H1>`, `` e ``, etc. O protocolo HTTP só especifica que o texto de um arquivo até o seu término, no caso de uma transferência de página - não valida o HTML do texto transmitido. Por outro lado, o programa cliente, ao encontrar um comando desconhecido de HTML deve ignorá-lo. Não deixar de mostrar o restante do documento por causa de um comando incorreto, mas simplesmente assumir que é um comando que ele desconhece, e ignorá-lo, continuando a mostrar o documento HTML como se nada tivesse acontecido.

Desta forma, Andreessen pode criar o seu Mosaic interpretando o novo comando `` do HTML que permitia a inserção de uma imagem no texto. Os documentos criados que fizessem uso desse comando poderiam ser colocados nos servidores já existentes. E se fossem acessados com um browser antigo, ou outro sem suporte a esse comando ``, o documento seria exibido normalmente - sem a imagem.

A inovação da adição de imagens foi bem-vinda, e as demonstrações da World Wide Web do próprio CERN eram feitas usando o Viola. É difícil nos dias de hoje

⁴http://www.scholars.nus.edu.sg/writing/ccwp10/rina/new_browsers.html, acessado em 16/11/2002

⁵<http://www.dejavu.org/mosaic-pres.htm>, acessado em 15/11/2002

tentar rastrear como a linguagem HTML foi se expandindo nesses primeiros meses. Como muitas coisas em informática, a especificação do HTML inclui um número de versão - então, o HTML oficial, que era sancionado por Berners-Lee incluiria em suas novas versões alguns dos comandos que teriam surgido nos programas de navegação escritos independentemente.

Eventualmente Tim Berners Lee se desligou do CERN e criou o W3C - World Wide Web Consortium, uma organização que visa justamente dar os parâmetros não só para o padrão HTML, mas para uma série de outros protocolos utilizados na Internet, que desta forma funcionam de forma aberta, e independente do programa que as implementa.

De qualquer forma, enquanto a World Wide Web se popularizava nos idos de 1993, e principalmente 1994, ano em que se iniciou o acesso à Internet a partir de provedores privados nos Estados Unidos, os produtores dos navegadores foram acrescentando novas extensões ao HTML.

Uma característica que logo se sentiu falta foi a habilidade de se especificar a cor e o tipo de fonte de texto, e logo, lá estava, o suporte ao comando `font` que permitia que fosse especificada a fonte, cor, tamanho, e outras características de um trecho de texto.

Com o surgimento do navegador Internet Explorer, em sua versão 3.0, houve praticamente uma guerra declarada entre a Microsoft e a Netscape - produtora do navegador mais popular até então. Nessa guerra, uma das armas era a inclusão de mais e mais extensões ao HTML, de forma que cada navegador pudesse oferecer uma experiência visual mais rica. Como resultado dessa guerra o HTML foi crescendo

de forma quase descontrolada. Comandos aceitos em um navegador eram ignorados em outro.

A situação chegou a um ponto em que a W3C parou de incluir muitas das extensões surgidas desta forma na especificação do HTML, e até mesmo a desestimular o uso de muitas das que já haviam sido incluídas. Não sem antes editar a recomendação de uma linguagem separada, complementar ao HTML chamada de CSS (Cascading Style Sheets).

Com o mote de separar conteúdo e apresentação, a idéia da introdução do CSS foi deixar o HTML marcar a estrutura do texto: as sessões, parágrafos, títulos, listas e tabelas, e o CSS marcar como esse texto é apresentado na tela: posição, cor, tamanho da fonte, etc...

Sendo especificamente projetada para isso, o CSS introduziu uma nova flexibilidade às páginas WEB ao mesmo tempo em que tornava o HTML mais simples novamente. Isso foi um passo importante, já que o HTML, sendo texto, sempre foi uma linguagem que tinha a intenção de poder ser lida e escrita diretamente por pessoas, que podem ou escrever suas páginas na WEB diretamente nessa linguagem, ou realizarem ajustes finos em páginas que estejam fazendo com auxílio de programas que gerem HTML automaticamente. Com o advento de todo o tipo de estilos e especificação de fontes e cores embutido no HTML, os comandos se tornaram mais complexos, maiores e menos legíveis, ao ponto de em alguns casos, o conteúdo textual se perder em meio aos vários comandos de um arquivo HTML.

Com a separação introduzida pela CSS o HTML se tornou novamente uma linguagem que pode ser facilmente alterada diretamente pelo autor, sem uso de

outros programas, e mais ainda, uma vez que a CSS também é uma linguagem desenhada para ser editável manualmente, a adição de estilos e layouts à páginas na Internet foi enormemente facilitada.

Entretanto, são poucas pessoas hoje em dia que editam manualmente suas próprias páginas. Por que por outro lado, o conceito de “computador caixa preta” chegou também às páginas de HTML, e, frequentemente as pessoas usam editores que permitem que se escolham as posições de texto, tipos de letras, imagens, com cliques no mouse, e sequer sabem que bastaria abrir o mesmo arquivo que salvam a partir desses programas em outros editores de texto para poderem ter acesso direto aos comandos do HTML.

E claro que, se tentassem fazê-lo, sem antes se prepararem, as chances são mais de que se assustariam do que de que achassem algo interessante, uma vez que o HTML, apesar de ser uma linguagem simples, frequentemente é criado por programas que geram seu código automaticamente se torna de difícil legibilidade. O princípio é simples: se o conteúdo do arquivo vai ser acessado apenas de forma automática, quer por um editor, quer por um navegador que vá interpretar o HTML, então o código não precisa mais ser facilmente legível por humanos. Pode-se, ao contrário, priorizar um código que seja mais simples de ser gerado automaticamente, embora mais extenso e mais complexo.

De fato, no início da WEB, dentro do próprio CERN, e depois aos poucos pelo mundo, a idéia sempre foi que as pessoas editassem suas próprias páginas manualmente. Com isso, além de resolver os problemas de comunicação do CERN, Berners-Lee proporcionou a criação do que poderia ser a Biblioteca Mundial, um espaço em que todos pudessem escrever suas idéias, e ligar, de forma simples, suas

idéias as de outras pessoas. Uma grande teia de pensamentos, que inevitavelmente nos traz a mente o funcionamento de nosso próprio cérebro, com unidades que se entreconectam para formar, a partir de estruturas simples, o pensamento humano.

Será então que esses milhares de textos escritos e entrelaçados nessa teia de alcance mundial formariam uma espécie de supermente? O quanto próximo do processo do próprio pensamento o ato de “surfear na web” estaria? Com efeito, os que tinham acesso à internet nessa época tiveram uma experiência única do momento, em que navegar na web era ler uma série de textos escritos por pessoas comuns, e os textos relacionados entre si ligados uns aos outros pelos links de hipertexto.

Mas a natureza da Web foi se alterando. A vontade de escrever das pessoas podia ser bastante, mas como achar uma página sobre um determinado assunto? Surgiram os “portais- sites destinados a concentrar a informação para a localização de outros sites - e dentro deles os robôs de busca - programas automáticos que liam a Web co-relacionando palavras chaves em bancos de dados para permitir a procura à páginas pelas mesmas.

Por outro lado, as indústrias e grandes empresas começaram a ter seus sites - esses não mais estariam restritos a ser um ou mais textos escritos por uma pessoa, ligados ao restante da web. Precisavam ser uma cara da empresa - a presença da empresa na Internet. Mais ainda - como ter lucro com essa presença na Internet?

Esses fatores, aliados com o acesso à internet por pessoas que começavam a ver o computador apenas como uma caixa preta, algo de se ver, de se assistir como uma TV, levaram a profundas transformações na Web. No ponto em que a mesma se encontra hoje, a maior parte das pessoas que utiliza a internet tende a ver alguém

que “sabe fazer home pages” com respeito, como se fosse um saber distante, quase de outro mundo.

Os links de hipertextos, idealizados originalmente para ligarem idéias dentro de textos a textos que tratassem das mesmas, do mesmo autor, ou de outro em qualquer parte do mundo, se tornaram não mais que meios de se acessar o conteúdo a partir de um índice, composto por editores, apontando para notícias dentro de um mesmo site, controles para adicionar um produto à cesta de compras, e assim por diante.

Chegamos ao ponto em que, apesar de a proposta original da World Wide Web ter sido a de ter todos os documentos diretamente acessáveis através de um endereço único para cada texto, onde teríamos tantos escritores quanto leitores, temos uma reprodução das formas tradicionais de escrita na mídia - poucos escrevem para muitos.

No entanto, há exceções. De um lado, muitos sites na Internet oferecem hospedagem gratuita de páginas pessoais, mantidas financeiramente pela inclusão obrigatória de um anúncio em cada página. Esses sites em geral, além da hospedagem, oferecem também “ferramentas” de criação de sites, na forma de modelos prontos, em que o usuário final praticamente só escreve o seu texto dentro de um design pré-definido.

De outro, há variações dessas páginas pessoais, como os chamados “blogs- diários eletrônicos automatizados. Mais simples de atualizar e manter que uma página HTML pura, a idéia é que o dono de um “blog- contração de “web” e “log- descreva periodicamente suas atividades, de forma que as mesmas possam ser acompanhadas por amigos. Na prática, é uma página pessoal em que, após se identificar com sua senha, o dono do blog possa simplesmente digitar o texto que pretende acrescentar à

página naquele momento num formulário do site, e ele é automaticamente acrescentado a página. Essa simplificação atraiu muito do público que se sentia intimidado pela aparente complexidade de “fazer uma página”- no entanto, há uma tendência maior de que Blogs conttenham apenas dados cotidianos, referentes ao dia a dia, e relevantes ao círculo social do autor. Em contraste, home pages pessoais tendem a agregar informações sobre o interesse ou os hobbies dos autores, atuando diretamente como fontes de informação que compõem a Web. Exemplos muito comuns são páginas pessoais com letras de música, ou sobre fotografia.

4 O que já Existe para o Currículo Hoje

4.1 Introdução

No capítulo 1 foram feitos vários apelos ao computador como um instrumento criativo, algo muito além de uma máquina de escrever. Aqui serão citadas algumas das múltiplas ciências e artes cuja pesquisa e criação se tornam, senão facilitadas pelo microcomputador doméstico, ao menos abordadas sob aspectos inteiramente novos. E em muitos casos, há a possibilidade de trabalho - entendido também como pesquisa e criação - em campos que antes exigiam equipamento especializado de alto custo.

Lembrando sempre que este trabalho visa apenas informar algumas das possibilidades em cada campo, quando houver, descrever em linhas gerais as potencialidades de alguns programas disponíveis. Para um trabalho efetivo de elaboração de plano de aulas, ter-se ia que experimentar com vários software de cada área, afim de estabelecer as atividades a serem trabalhadas com e pelos alunos, conforme o caso.

4.2 Matemática

Como no currículo escolar as disciplinas mais importantes, ao menos em termos de carga horária, são justamente a comunicação e expressão e a matemática, esta, logo depois da escrita, talvez seja entre as disciplinas curriculares a que tem uma maior possibilidade de exploração no âmbito da informática sem uma pesquisa extensiva.

De fato, para cálculos puramente numéricos - em particular as 4 operações que usamos no dia a dia, ensinadas desde a primeira metade do ensino fundamental - o uso de aparelhos baseados na eletrônica digital se popularizou vários anos antes do que ocorreu com a escrita.

Como visto, pode-se situar a popularização da escrita em plataformas digitais num período de até 10 anos atrás (1993). Enquanto que para as 4 operações básicas, é comum o uso de calculadoras digitais com display de cristal líquido desde o início dos anos 80.

De fato essas calculadoras se baratearam tanto, e têm, em sua maioria uma interface tão simples, que convivem - e são usadas - lado a lado com microcomputadores que literalmente podem executar algumas das operações realizadas nas calculadoras mais de 3 bilhões de vezes num único segundo.

Mas essa matemática mais rápida dos computadores é limitada. Justamente limitada a computações numéricas mais simples. No mundo digital, a matemática se divide em duas categorias estritamente diferentes uma da outra. Por um lado, operações numéricas operando sob uma faixa limitada de números, por grande que

seja esse limite. Por outro, a matemática simbólica, algébrica - que não depende. A primeira é intimamente associada aos princípios básicos de funcionamento do computador. A segunda é tão distante desses que só é possível através de programas especializados em matemática simbólica, dos quais o mais popular é o Mathematica - um programa proprietário da autoria de Stephen Wolfran. Há alternativas livres, como o pacote GNU Jacal.

No campo numérico, tem-se que continuar o cuidado - que a escola já adota - de não substituir o aprendizado das 4 operações básicas, e de outras operações numéricas, pela simples execução dessas mesmas operações num computador ou numa calculadora. Mais do que proficiência nas mesmas, é através delas que se pode chegar a compreensão da matemática simbólica - sem a qual mesmo o melhor dos programas de cálculo simbólico seria apenas uma sequência de símbolos sem nexo.

Uma categoria de programa interessante para manipulação numérica, é a planilha de cálculo. Um tipo de programa que surgiu como o Visicalc na década de 80, hoje é mais conhecido através do programa proprietário Microsoft Excel, presente em muitos computadores em que sequer é usado já que é oferecido em venda casada com o quase onipresente Microsoft Word. Mas o Excel é apenas uma das opções de programas de planilha existentes, havendo várias opções com a mesma funcionalidade em software livre: o Kspread, parte do KOffice, o módulo de planilha do Open Office, e o Gnumeric.

As planilhas de cálculo tem como característica predominante de sua interface uma enorme tabela - a planilha em si - onde em cada célula pode ser colocado texto, um número, ou uma fórmula. O que a torna diferente de uma planilha em papel é que uma fórmula colocada é processada automaticamente, e o seu valor é

imediatamente visualizado. Assim, pode se entrar com o texto “=2*3 + 10” numa célula qualquer, e ver seu valor mudar instantâneamente para “16”. Sua verdadeira utilidade surge quando se digitam nas fórmulas além de números fixos, referências a outras células. Assim, quando essas células são atualizadas, a célula que contém a referência também tem seu valor alterado. Assim, uma das funções mais usadas no cotidiano extra escolar é a soma de uma faixa de células que totaliza os valores de uma tabela.

Se por um lado é difícil imaginar como uma tal classe de programa possa funcionar com a matemática escolar tradicional, que tenta funcionar independente das outras disciplinas ou do mundo, a simples elaboração dos populares “problemas de matemática- situações cotidianas envolvendo números que em geral requerem um sistema de equações para sua resolução - pode resultar em uma gratificante produção para o aluno - que seria responsável por identificar os dados necessários para o seu problema, talvez coletar esses dados através de pesquisa na internet, questionários, bibliotecas - e resolver que operações realizar sobre seu conjunto de dados.

Os programas de planilha em ambientes gráficos ainda permitem uma elaborada sofisticação na apresentação final do trabalho. Dependendo do tema escolhido, a reinvenção de um simples problema de matemática pode se transformar num trabalho prazeroso, onde o aluno entra em contato com o significado dos números, mais que com as operações em si, que tem como resultado final algo que não fique nada a dever ao “mundo adulto”.

Um exemplo de trabalho nesta linha poderia ser: Pedro, que sempre foi um bom desenhista, indignado com a guerra resolve fazer uma camiseta clamando pela paz no mundo. Faz um desenho no computador, e o leva até uma gráfica rápida que

cobra R\$10,00 por uma camiseta branca mais R\$4,00 pela impressão na frente da mesma. Os amigos de Pedro gostam da camiseta e encomendam camisetas parecidas. Sendo os preços para camisetas coloridas R\$15,00 e sua impressão R\$8,00, e ainda a opção de manga comprida por mais R\$5,00. Supondo que haja 40 pessoas na classe de Pedro, elabore uma planilha de cálculo em que Pedro possa digitar suas encomendas, de forma a visualizar rapidamente de quanto dinheiro precisará para toda a empreitada. Preencha a planilha com valores que considere razoáveis.

Além de outros, combinados com dados geográficos e demográficos, por exemplo, que poderiam ou ser fornecidos junto com o problema, ou deixados a cargo do aluno conseguir. É claro também, problemas inversos: planilhas entregues no formato digital aos alunos, contendo dados reais e dados absurdos - que deveriam ser analisados pelos alunos.

Exemplo: Uma planilha com uma lista de nações, sua área total em quilômetros quadrados, seu número de habitantes, e duas colunas calculadas: uma com o número de habitantes por quilômetro quadrado - a densidade demográfica, e outra, derivada desta, com a "densidade demográfica por quilômetro quadrado- um número sem sentido algum. Ficaria a cargo dos alunos descobrir a irrelevância dos valores nessa última coluna.

No aspecto simbólico da matemática, apesar de haver programas específicos, a abordagem mais apropriada para o ensino fundamental se daria através de alguma linguagem de programação. Os pontos de partida para isso serão vistos no próximo capítulo.

4.3 Biologia

No campo da Biologia, a informática anda em alta por conta das pesquisas em torno de sequenciamento de DNA e outros estudos com computação de alto desempenho. Embora os computadores pessoais de hoje possuam uma capacidade de processamento tal que não sejam negligíveis perto dos equipamentos onde são feitas as simulações para a pesquisa nessa área, a pesquisa em si, a captação original dos dados, seu tratamento, exigem conhecimento altamente especializado, além de laboratórios.

Portanto, por mais que ocupe a mídia, esse tipo de abordagem da biologia através da informática não é adaptável ao currículo escolar. Há bastante campo para agregação do estudo da biologia à informática com programas do tipo enciclopédias multimídia, e mesmo na Internet, como fonte de pesquisa e espaço para exposição dos trabalhos feitos. A fixação do conteúdo, mesmo que ensinado da maneira tradicional, pode se dar também através de jogos de trivia ou outros. Um simulador livre de ecologia - no estilo do clássico SimCity seria um jogo excepcional.

De fato, no início dos anos 90 a mesma empresa que faz o SimCity chegou a fazer o SimEarth e o SimLife um simulador de ecologia planetária e um de evolução de espécies, além do SimAnt. Esses títulos hoje encontram-se fora de catálogo, e não podem ser adquiridos legalmente. Relembrando o que foi visto no capítulo II, não seria difícil atualizar esses jogos para os ambientes dos sistemas operacionais atuais se o código dos mesmos fosse livre.

Um simulador de genética mendelviana, no entanto, seria um programa simples de ser criado num projeto mensal, pelos próprios alunos, sendo uma sugestão para

emprego da programação como será abordada no capítulo 5. Planilhas eletrônicas também podem ser feitas para estudos de população, e da relação entre as mesmas.

4.4 Geografia

Com o uso de planilhas eletrônicas, como as descritas na sessão de matemática, torna-se relativamente fácil a elaboração de várias simulações, a partir de dados demográficos, ou mesmo geológicos. Com isso, a geografia pode ganhar uma alavanca que permite que seu ensino seja transformado de um que se dá normalmente de forma passiva - com os vários dados sendo apresentados aos alunos, em um ensino que permita a prática ativa investigativa. Como varia a densidade demográfica de áreas rurais em relação a disponibilidade de água? Uma vez estimado isso, pode-se estimar quantas pessoas poderiam ocupar uma região após um projeto de irrigação. Em quanto tempo estarão exauridas as reservas atuais de petróleo? E se forem descobertas novas reservas equivalentes a 20% das já conhecidas?

O já citado jogo SimCity aparece também quase como uma necessidade no estudo da geografia. Embora seja um software proprietário, há similares livres do SimCity clássico - um simulador de cidades onde pode se acompanhar a qualidade de vida de uma pequena cidade, conforme se fazem investimentos em educação ou no trânsito, por exemplo. Em escalas maiores, e se integrando ao ensino de história, podemos citar o simulador Civilization, e jogos similares, incluindo o livre FreeCiv.

Como a biologia, a geografia também se beneficia da Web e de enciclopédias multimídia como fonte de conteúdo.

No campo da astronomia, por vezes integrada na disciplina geografia nos currículos, a informática oferece várias possibilidades, principalmente com simuladores de posição dos planetas, que funcionam como os tradicionais planetários.

4.5 Língua Estrangeira

Senão por mais coisas, a Internet permite o acesso direto a textos reais - isso é, não simplificados - em qualquer língua falada por uma população significativa. Nada de Ivo viu a Uva, qualquer pessoa pode ler as notícias do dia em Francês ou Italiano, ou Mandarim se tiver acesso à internet.

Além disso, como muitos um dos métodos de ensino de línguas mais difundido é justamente a resolução de exercícios repetitivos, que acabam reforçando o aprendizado por meio de retorno (feedback) positivo a respostas corretas, torna-se simples uma adaptação desse método para o meio digital, e com retorno muito mais rápido para o aprendiz - o que é algo que aumenta a eficácia desse método de ensino.

Além da leitura de textos, também é possível conversar - nas modalidades já descritas no capítulo 3 - com falantes de qualquer língua. Por enquanto essa “conversa” é mais prática na forma de mensagens de texto instantâneas, mas a conversa por voz também é possível e deve se popularizar mais ao longo dos próximos anos.

4.6 Química

Uma busca rápida pela Internet revela dezenas de programas, licenciados como Software Livre, que são na verdade ferramentas para a química. As descrições vão desde programas para visualização tridimensional de moléculas, passando por simuladores de complicadas moléculas biológicas, até ferramentas para escrita e gerenciamento de equações químicas.

A maior parte desses programas no entanto, é pensada para uso como ferramenta “de produção” científica, para uso em situações reais, e não escrita com fins didáticos. Mas justamente essa aproximação do “mundo das disciplinas escolares” com o “mundo real”, através do acesso provido ao estudante às mesmas ferramentas usadas no trabalho profissional é que faz da informática um diferencial para ensino.

Como um dos programas mais amigáveis e completos, destaca-se o *ghemical*¹, em que se pode simplesmente desenhar a molécula desejada, átomo por átomo.

Pode-se dispor do computador como uma poderosa “calculadora química” com programas tão amplamente divulgados quanto as planilhas de cálculo descritas acima, na sessão que trata de matemática, ou com programas específicos para essa finalidade.

¹<http://bioinformatics.org/ghemical/>, acessado em 20/04/2003

4.7 Física

Novamente, como na química, centenas de simuladores dos mais diversos assuntos dentro da física se encontram disponíveis.

É interessante notar que justamente por lidar com modelos simplificados dos assuntos de que trata, torna-se relativamente simples escrever pequenos simuladores de cunho próprio para problemas específicos de física. O próximo capítulo abordará esse aspecto do uso da informática.

Um programa de destaque é o jogo - infelizmente não livre - chamado The Incredible Machine. Uma versão livre do mesmo, chamada The Penguin Machine está nos estágios iniciais de desenvolvimento. Mostra alguns princípios intuitivos da física de forma lúdica, bem adequado para familiarização com os conceitos de física, mais que com fórmulas.

Mesmo uma planilha eletrônica pode ajudar a perceber rapidamente as relações entre posição, velocidade e aceleração, através de gráficos e/ou valores que se ajustem automaticamente quando outros são modificados. Por exemplo, as coordenadas de um corpo em estudo podem se modificar conforme sejam digitados valores diferentes para aceleração.

Ademais, no capítulo 5 está incluso um exemplo de como a programação e o estudo de física podem ser abordados em conjunto.

No campo da ótica, o excelente software POVray, sobre o qual falaremos mais, pode ser usado para simular lentes com qualquer índice de refração e curvatura,e

estudar suas combinações com outras.

E por fim, estão disponíveis também vários programas para uso em “pesquisas reais” que podem ser usados num ambiente de ensino, para se verificar a resposta de experimentos, simulações numéricas de problemas complexos, e outros.

A eletrônica, que é incluída na disciplina de física nos currículos escolares, tem muito a ganhar quando empregada em conjunto com a informática, tanto para ensino quanto em aplicações reais.

E o motivo é bem simples: pode-se simular o funcionamento de qualquer tipo de circuito, sem que se tenha acesso aos componentes eletrônicos em si. Além disso, as mesmas simulações permitem que se efetuem medidas de grandezas que normalmente permaneceriam ocultas.

Num projeto realista, se fosse necessária a alteração de um componente para melhorar o circuito, ter-se ia que adquirir o substituto para então se efetuarem os testes. Com o auxílio de um simulador, pode-se escolher o componente mais adequado para todo o sistema, para então se proceder à compra, para a montagem final do circuito. Como software livre, destaca-se o programa “ng spice”, nessa categoria de simuladores.

4.8 Fotografia e Artes Plásticas

Fotografia, eis um campo em que a informática esta operando modificações tão significativas quanto as que já se operaram na escrita. Estamos em plena na subs-

tituição da fotografia tradicional - sobre filme químico, para a fotografia digital, em que as imagens são armazenadas como coleções de dados, tais como as que já eram usadas em computadores.

Se por um lado as máquinas fotográficas digitais se mantêm como dispositivos separados do computador pessoal em si, por outro lado, a capacidade de aplicar efeitos, editar ou transformar uma fotografia faz parte do universo da computação pessoal.

Como no caso da escrita no entanto, por questões de marketing, e outras, de patentes, os mais conhecidos programas para edição de fotografias e desenho a mão livre no computador, são software proprietário. Sendo um dos poucos campos em que a Microsoft não tem monopólio no mercado, os programas proprietários mais utilizados na área são o Adobe Photoshop e o Corel Photopaint.

Há no entanto uma consagrada opção Software Livre para edição de imagens - o GNU Image Manipulation Program - GIMP. Implementando quase toda a funcionalidade desses outros programas, exceto nas áreas em que a operação dos mesmos é protegida por patentes. Patentes, como as já discutidas, que efetivamente proíbem algumas técnicas de seleção de cores sem que se paguem royalties para essa ou aquela empresa, que tornam a funcionalidade do GIMP um pouco reduzida para, por exemplo, o mercado publicitário profissional.

A parte dessas limitações artificialmente impostas por uma cultura que insiste em privatizar, não só o conhecimento, como agora as próprias cores e a matemática, o GIMP se presta muito bem a edição de imagens, sendo que transformações que seriam muito difíceis, senão impossíveis com técnicas de laboratório para a fotografia

tradicional podem ser feitas em um piscar de olhos. Por exemplo, aumentar o contraste de uma fotografia já tirada, ou substituir uma cor por outra na foto.

Virtualmente todas as imagens impressas com que temos contato no dia a dia, quer criadas no computador, quer fotografias em publicações de papel, passam por um programa de manipulação de imagens desse tipo. Uma aula de design gráfico ganha dinamismo e potencial, quando pode exemplificar o que se ganha ao avivar as cores, ou aplicar-se uma sombra falsa a um texto numa peça gráfica.

E as ferramentas disponíveis para uso na escola são as mesmíssimas que se usa em qualquer estúdio.

Abaixo, um exemplo de como uma fotografia pode ser reduzida a duas cores, sem meios tons. Um efeito que com um laboratório convencional só é possível através de um uso intenso de filtros, por vezes caros:



Figura 1: Contraste normal



Figura 2: Alto contraste

Ao tratar das outras artes-plásticas, recaímos principalmente na pintura e desenho, que podem se beneficiar dos mesmos programas utilizados na edição de fotografias para a criação de novas imagens. Desde o primeiro computador pessoal vendido com uma interface gráfica baseada no Mouse, o Apple Macintosh, de 1984, já veio incluída uma ferramenta de desenho com o mesmo, que inclui uma “paleta de

ferramentas- uma paleta estilizada, derivada da paleta de tintas do pintor. Em vez de selecionar uma cor, o “artista digital” escolhe um tipo de pincel, ou a borracha, ou o spray, e com isso muda o tipo de marca deixada pelo botão pressionado do mouse sobre a “tela digital”. Quando essa interface se popularizou, com o windows, esse programa, o Macpaint, foi clonado e se encontra com o nome de “Paint” nas versões mais novas do windows. Como software livre, temos o xpaint, e também o GIMP para suprir essas funcionalidades.

Esses programas permitem, para o artista sério, sobretudo a possibilidade de retroceder “pinceladas”, ou de combinar cores ou objetos separadamente na mesma imagem, podendo alterar qualquer um deles, sem alterar toda a imagem.

Já uma coisa que sempre fascinou a todos, desde que os computadores começaram a se tornar populares, são os desenhos “3D- isso é, projeções na tela de modelos tridimensionais de objetos gerados por computador. Um programa clássico nessa categoria é o POVRay - Persistence of Vision Raytracing Utility. O nome do programa homenageia o quadro de Dali “Persistence of Memory”, e de fato ele é capaz de executar imagens bem próximas as daquele quadro - sempre dependendo da capacidade do artista que o usa.

O POVRay no entanto se caracteriza por não ter uma interface gráfica. A cena que se quer desenhar é descrita numa linguagem própria do mesmo. Isso em alguns casos facilita a colocação de objetos em posições exatas. A cultura do “Computador como ferro de passar roupa” no entanto tende a acostumar as pessoas com programas em que tudo funciona apertando-se apenas um botão. Isso é nocivo, como já foi visto, e, tendo tido contato com a computação pessoal dentro dessa ideologia, a impressão é que programas como o POVRay são mais complicados do que parecem. Mas o

fato é que para construir uma cena foto-realística do nada, não se pode preceder com um simples arrastar de mouse e um único botão. Melhor dizendo, até se pode, mas certamente se toma mais tempo assim, do que num modelo em que se possa colocar por escrito exatamente o que se queria, e reaproveitar trechos do que já se viu escrito, quando alguns atributos dos objetos a serem desenhados se repetirem.



Figura 3: Exemplo de imagem criada com o POVRay

As ferramentas de desenho e edição de imagens nos computadores, ao contrário de muitas outras, são por natureza fáceis de serem utilizadas, e não há empecilhos para que aulas com o uso dessas ferramentas não sejam intercaladas com aulas em que a criação se de com o lápis de cor, ou com tinta.

4.9 Música e som

As interações da música com a eletrônica começaram ainda antes da era digital, com equipamentos analógicos construídos para acrescentar efeitos especiais a sons na forma de sinais elétricos, ou mesmo brincadeiras de recortar e colar fitas magnéticas, que permitiam que o som fosse manipulado como se fosse um objeto físico.

Com o advento de computadores pessoais, com poder o suficiente para tanto, essas ferramentas também foram transferidas para o ambiente digital, e há programas para se acrescentar efeitos, emendar e remendar músicas ou sons previamente gravados. O som é visualizado através de sua onda sonora, que é representada como um gráfico na tela, na maior parte desses programas. Por exemplo o software livre Audacity permite saída em vários canais de som, com qualidade de estúdio.

Outra categoria de programas é a de edição de música. Por vezes numa interface que simula a pauta usada para escrever partituras, o usuário pode posicionar as notas diretamente na mesma, ou ainda, tocar a música num teclado eletrônico conectado ao computador, e ver a partitura surgir na tela. Com a partitura pronta, nesse formato eletrônico, é possível ouvi-la sendo tocada por outro instrumento - simulado pelo computador. Ou por vários. Pode se copiar as linhas melódicas, e modificar as cópias para que se tornem arranjos para outros instrumentos.

Dessa forma, sempre com trabalho árduo por parte do músico, afinal, não há magia, há apenas facilidades - um músico pode escrever um arranjo para toda uma orquestra, e ouvir o resultado final, sem precisar reunir a orquestra em si. O Software livre tem no Muse - Music Editor um grande programa que permite esse tipo de trabalho.

Uma outra categoria de programas são os sintetizadores, que permitem a criação de novos tipos de som, novos “instrumentos”, a partir de fórmulas matemáticas. Nessa categoria há o imbatível CSound, que inclui até uma linguagem de programação própria para a especificação da forma da onda dos instrumentos.

Obviamente, o ensino de música, quando houver, deve se dar prioritariamente através de instrumentos musicais e do canto. O computador pode se fazer presente como uma ferramenta a mais principalmente para os alunos que se propuserem a estudar mais sobre a natureza do som, ou dos acordes musicais. Para tanto, se prestam programas como o CSound.

Eventualmente atividades exploratórias, em programas de edição de partituras, como o MusE poderão servir para os alunos experimentarem com vários instrumentos, ou a alteração de algumas notas em uma composição, a fim de poderem, mesmo sem ter a habilidade de tocar um instrumento, criarem coisas próprias sobre música já existente. A criação de peças musicais inteiras no entanto só poderia se dar a parte do tempo escolar, com muito estudo e dedicação, e certamente, seria mais fácil se fosse feita antes num instrumento convencional, e posteriormente transposta para o computador.

4.10 Projetos - Desenho Técnico

Novamente, uma área em que a informática se infiltrou e transformou os métodos de trabalho nos últimos 10 anos. Em todo o país, cursos de engenharia trocaram suas disciplinas de desenho técnico convencional, por desenho técnico com o uso de Software CAD- Computer Aided Design. De fato, o potencial para projetistas que

podem realizar o desenho de seu projeto uma única vez e depois vê-lo de todos os ângulos desejados é bem grande.

Nesse campo, predomina a ferramenta proprietária AutoCAD. Existem ferramentas livres, no entanto. Para desenho de projetos 2D apenas, existe o sólido QCAD, um programa fácil de usar, e bastante difundido. Para desenhos 3D há o lignumCAD e o Varkon CAD.

4.11 Vídeo

Conforme foi visto no capítulo 2, sobre o funcionamento do computador, não é difícil perceber que, no âmbito da informática, querer impedir que informações digitais sejam copiáveis, é quase como querer impedir a água de ser molhada. No entanto, os próprios discos de DVD trazem seus vídeos digitais criptografados de tal forma que não é possível que sejam reproduzidos - em qualquer sentido - sem chaves secretas -como senhas - em poder da indústria de vídeo. Essas chaves são distribuídas às indústrias que fazem aparelhos de DVD, e criam os “drivers” oficiais para reprodução de DVDs nos DVD-ROMs dos computadores pessoais. Como essas chaves não poderiam ser públicas, não haveria, em teoria, como reproduzir um DVD com um programa de Software Livre.

No entanto, no caso específico do DVD, as chaves foram quebradas, como já foi comentado, e hoje é possível se assistir um filme pelo qual o dono pagou, no aparelho pelo qual ele pagou, utilizando um programa que não comprometa sua própria liberdade. Exceto nos EUA, e em países que vêm clonando sua legislação chamada de DMCA - Digital Millennium Copyright Act, que especificamente proíbe

que se passe por cima de técnicas artificiais para proteger direitos autorais. Em outras palavras, é ilegal nos EUA decodificar um DVD com um programa do qual a pessoa possua o código fonte, mesmo que a intenção seja apenas de assistir ao conteúdo do mesmo.

Mas, apesar disso, há bons programas para se assistir a vídeo digital, entre os quais pode-se destacar o Mplayer, por sua capacidade de entender vários formatos de vídeo digital diferentes, dentre as centenas existentes.

Quanto a outros aspectos do vídeo digital, temos a edição de vídeo, que, se até bem pouco tempo atrás tinha que ser feita numa caríssima ilha de edição, pode ser feita hoje num computador doméstico equipado para tal. Para edição de vídeo ainda é necessário um pouco mais de memória RAM do que a que equipa a maior parte dos computadores em uso. Quanto ao software, uma das melhores opções de software livre é o Broadcast 2000, e, mais recentemente, o CINEPAINT - um programa derivado do GIMP citado acima, e adaptado para edição e retoque de vídeo digital.

Para criação de imagens digitais puras, com toque de “efeitos 3D”, pode se utilizar, sem nenhuma modificação, o POVRay, também já citado. Nesse caso, o fato das imagens serem criadas na forma de scripts, acaba se tornando uma vantagem razoável sobre programas com interface gráfica. Pelo simples fato de ser mais simples acompanhar o que acontece a cada instante com cada objeto, do que clicar-lo e arrasta-lo numa interface maluca, que tenta adaptar para quatro dimensões (as 3 espaciais, mas o tempo do filme) um dispositivo de entrada que opera em duas dimensões - o mouse.

4.12 Finalizando

Essas duas últimas categorias de software, como nas áreas anteriores, servem a um tempo para produção profissional, e também para o ambiente de ensino com bastante ênfase na criação dos próprios alunos.

Desta forma, mesmo sem cobrir de forma ostensiva todas as disciplinas, pode-se ter uma idéia do que existe disponível para ser trabalhado num ambiente de ensino que integre a informática.

5 Programação como Ferramenta de Criação

5.1 Programar - O Que É?

Chega-se então, a programação. O que é afinal, este palavrão? Algo mistificado entre os leigos ao ponto quase comparável de se chamar alguém de alquimista, ou bruxo, na Europa de séculos passados. Ninguém é queimado na fogueira por ser um programador, mas seus afazeres e estudos são tão obscuros quanto os daqueles.

Antes de entrar em detalhes sobre o que é ou deixa de ser a programação em si, convém considerar se há alguma validade na idéia de programação como parte do currículo. Para tanto deve-se ter uma idéia sobre a sua natureza, possibilidades, e o tipo de atividade à que corresponde.

Um programa nada mais é que uma série de instruções lógicas que o computador executará de forma geralmente sequencial, operando sobre dados arquivados, ou obtidos por meio da interação com o usuário. Todos os aplicativos que permitem fazer o que foi descrito no capítulo 4 são programas feitos por terceiros, e que o usuário final encontrará prontos para serem utilizados.

Será então necessário algo mais? Ora, se existem tanto aplicativos, com tantas finalidades, não haverá sempre algum que cumpra a tarefa necessária, sem exigir programação?

De certa forma a resposta é sim, haverá. Pelo menos, e felizmente, é assim para os aplicativos de maior porte, que exigiriam milhares - por vezes milhões - de linhas de código para serem funcionais. No entanto, estabelecendo ainda um outro paralelo com carros, fazer uso cotidiano de um computador, e não escrever pequenos e curtos programas seria o equivalente a ter um carro, e aproveitar sua capacidade de gerar eletricidade para a bateria, e a partir daí iluminação, aparelhagem de som, e até para alimentar um segundo carro elétrico, a partir do alternador do primeiro, mas sem usa-lo como meio de transporte.

Isso por que justamente a principal característica de um computador é poder seguir instruções lógicas em ordem, condicionadas por um ou vários fatores nos dados que está processando. E fazer isso repetidamente, de forma incansável. Esse ato, ordenar essas instruções para o computador é justamente programar.

Então tomemos um exemplo. Vamos supor que um usuário queira fazer um tratamento gráfico numa série de fotografias. Ele escolhe seu aplicativo favorito de tratamento de imagens, abre a primeira foto. Por tentativa e erro vai localizando uma série de tratamentos que deseja aplicar a todas as fotos a partir da primeira. Talvez aumentar o brilho e contraste, em seguida puxar a cor sépia, e aplicar um filtro para dar aspecto de rasgado e envelhecido às bordas da imagem. Todas operações que ele pode selecionar a partir dos menus gráficos do programa. Então, ele deseja fazer isso com outras 30 fotografias. Ele sempre tem a opção de mandar abrir uma por uma das imagens restantes, e aplicar os mesmos tratamentos, na mesma ordem,

a todas elas. Assim, ele pode utilizar o computador para realizar um trabalho repetitivo e mecânico e que, após a primeira imagem, não requer mais criatividade ou senso artístico.

No entanto, quase a totalidade dos aplicativos gráficos possui a capacidade de “scripting”, isso é o uso de scripts - que são na verdade pequenos programas - para justamente automatizar tarefas repetitivas. Então, nosso usuário que tenha conhecimento das capacidades efetivas do computador, pode, gastando um pouco mais de tempo para ler as instruções específicas de script do aplicativo que está usando, após executar as operações sobre a primeira foto, escrever um pequeno programa de 15 ou 20 linhas que não só a um único acionamento executará todas as transformações desejadas nas 30 fotos restantes - em uma fração muito pequena do tempo que isso levaria para se realizar manualmente - como ainda fará o mesmo com qualquer futura imagem à que se deseje aplicar o mesmo tratamento gráfico.

O exemplo pode parecer específico demais, mas pequenos programas de 20 linhas podem automatizar uma série vasta de tarefas que seriam repetitivas num computador, como enviar um conjunto de arquivos por e-mail para uma lista de pessoas, ou montar uma lista de execução apenas com as 20 músicas mais recentes que existam no computador. E são pequenas tarefas tão simples para o computador, mas que nenhum aplicativo existente fará de forma automática. Frequentemente pode-se encontrar tarefas que aparentemente levariam várias horas para serem executadas no computador, e que seriam realizadas em segundos por um script apropriado.

Programar é então a arte de conseguir dar ordens para o computador. E ao se entender seus princípios pode ao menos se saber quando é possível substituir uma tarefa tediosa e demorada, por outra, de programação, que ainda que tome uma

fração significativa do tempo da outra, não tem nada de tediosa, pois, programar é uma tarefa que se assemelha a resolver um problema de lógica, algo sempre desafiante para a mente humana, e na maior parte das vezes, divertida.

5.2 Dados Digitais

Um computador, como se sabe, é uma máquina. Apenas uma máquina, como um cortador de gramas, uma moto-serra, um carro ou um videocassete. Não executará quaisquer operações para as quais não tenha sido ajustado para fazer. A diferença para com as outras máquinas, é que ao invés de grama, arvores, gasolina, ou fitas de vídeo, um computador trabalha com dados. Mas que dados são esses? Começa de novo o misticismo..de onde vem? Na “vida real” encontram-se os tais “dados- mas parece que de forma esparça. Numa pesquisa eleitoral, no jornal, ou na previsão de temperaturas mínimas e máximas para o dia, no boletim meteorológico. Lá estão os números que alguém disse, ou se aprende que são, os tais dados.

Ora, ao se ligar o computador, tem se acesso a desenhos, músicas, textos, toda o tipo de coisas se movimentando, ou travando, de forma aparentemente aleatória. Onde estão os dados? A resposta é - em toda a parte. Tudo o que o computador faz, as imagens, sons, vídeos, são informações que em algum momento prévio foram codificadas em números. Digitalizadas.

Esse é o sentido do verbo mágico “digitalizar”. Codificar em números. A princípio pode parecer obscuro um processo que converta uma imagem em números. Segue-se um exemplo rápido, de como isso poderia ocorrer, de forma que se possa discorrer sobre os processos que se operam sobre esses números.

Tome-se uma foto 3x4cm, em preto e branco. Subdivida-se esta foto em elementos de imagem quadrados de 0,5mm cada um, tem-se então $60 \times 80 = 4800$ elementos de imagem na foto. Esses elementos de imagem são denominados pixels justamente do inglês "picture elements". Então começa-se a elaborar nossa lista de números. Tem que se ter, por exemplo, previamente convencionado que nossa lista de números conterà, no primeiro número, o tipo de colorido da imagem. Convencionamos o zero (0) para preto e branco, um (1) para colorido. O segundo número sendo a quantidade de elementos de imagem em cada linha da figura, o terceiro, a quantidade de linhas de nossa figura, e do quarto em diante, a intensidade da cor de cada elemento de imagem, da esquerda para a direita, e do topo para baixo. Essa intensidade de cor pode ser convencionada atribuindo-se o número cem a um pixel branco, e zero a um pixel preto, por exemplo. Está feito. Podemos agora ter uma lista de 4803 números que representa uma foto 3x4, como na figura 4.



Figura 4: Foto Digitalizada

Então, quando um computador exibe uma imagem, ele está de fato interpretando uma lista de números semelhante a essa do exemplo, e gerando outros números, que são enviados de forma sincronizada para o monitor de vídeo, ou para a impressora, onde através de um processo físico, são finalmente projetados na forma visível. Há

um fato crucial neste exemplo que pode ter escapado ao leitor. Foi dito "...tem que se ter (...) previamente convencionado ...". E isso de fato é o que faz os computadores funcionarem. No caso do exemplo, a única maneira da foto ser reproduzível é que haja uma convenção que especifique o que significa cada número. Os três primeiros números, fixos em cada imagem, são normalmente chamados de cabeçalho da imagem. Os restantes são a parte de dados. Suponha-se por exemplo que não se tivesse respeitado a convenção, e se tivesse trocado o segundo e o terceiro número do cabeçalho - o programa que efetua a leitura tentando ler uma imagem com 80 elementos de imagem por linha - teríamos algo como na figura 5. Essas convenções podem ser padronizadas, e abertas para quem desejar escrever um programa capaz de interpretar dados naquele formato, ou podem ser mantidos em sigilo, como um segredo de uma empresa específica de software, caso em que outras pessoas não serão capazes de escrever programas que interpretem aqueles dados, exceto por tentativas, erros, e experiências utilizando-se um conjunto de dados.

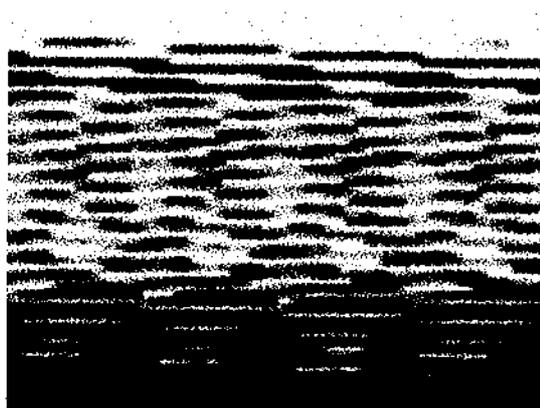


Figura 5: Leitura Incorreta de Foto Digitalizada

Ainda no caso do exemplo acima, pode-se notar que a foto 4 ficou falhada - não parece uma fotografia real, esta cheia de quadradinhos, o contraste entre os mesmos está grande. Isso acontece porque escolhemos tomar unidades de imagem relativamente grandes, de meio milímetro, e escolhemos um número relativamente pequeno de cores. Este número mágico de "elementos de imagem" que no nosso caso

é de 4800 pixels, é utilizado correntemente como “número mágico” de marketing pelos fabricantes de câmeras fotográficas digitais. É comum se ver em anúncios câmeras de 2.1 megapixels, ou 5 megapixels. O exemplo acima é de uma foto de 4.8 kilopixels. Ou seja - nesse caso trata-se realmente de um número que - via de regra - quanto maior melhor, o escolhido pelos fabricantes. Essa informação sozinha no entanto diz pouco sobre as características óticas das câmeras.

5.3 O Funcionamento do Computador

Mas, de volta a programação. Agora que se entende como as imagens são digitalizadas, e se sabendo que para sons, filmes, e textos, temos processos semelhantes, torna-se mais fácil compreender que o que um computador faz então, quando se diz que ele está “processando dados”, é executar uma série de operações lógicas e matemáticas sobre uma ou mais sequências de números. Essas operações são especificadas no que se chama de “programa”. O que é um programa então? Nada senão outra sequência de números. Tão simples quanto isso. A diferença é que é uma sequência de números interpretada de forma diferente. Os números que compõe um programa de computador são convencionados de forma que cada número represente uma operação lógica, seja ela uma cópia de um dado de uma região para outra da memória do computador, uma operação matemática entre dois ou mais dados na memória, ou um salto para outra região do próprio programa, condicionado ou não por estados anteriores dos dados.

Que operação lógica corresponde a que número é algo que é convencionado pela empresa que fabricou o microprocessador do computador. Se esta empresa optasse por não divulgar o funcionamento lógico, e que código numérico corresponde a qual

operação do microprocessador, ninguém, mesmo nenhuma outra empresa, no mundo, poderia escrever um programa para aquele microprocessador específico.

Tudo o que um computador faz então é executar programas. No momento em que é ligado, o microprocessador acessa uma área especial na memória do computador conhecida como ROM (Read Only Memory - memória apenas de leitura) - ou BIOS nesses dias de computadores compatíveis com os IBM PC - , que é uma área da memória onde não se pode armazenar dados, apenas ler, e apenas os dados ali colocados pelo fabricante do computador. esta região contém um programa que executa alguns testes no computador, e carrega para outra região da memória um programa que esteja num diskette, CD-ROM ou no disco rígido. Esse outro programa em geral é um pequeno código responsável por carregar o Sistema Operacional do computador. O Sistema Operacional, por sua vez, é o programa que está sempre em execução nos computadores modernos depois de ligados. Ele providência uma série de serviços básicos para o uso do computador pelos outros programas e inclusive, na maioria das vezes, por chamar os outros programas necessários para comunicação do usuário com o computador.

Em geral o sistema operacional disponibiliza, através de algum outro programa integrado ao mesmo, um ambiente gráfico onde se mostram menus e ícones de outros programas que serão invocados pelo usuário. Além disso ele disponibiliza toda uma série de serviços básicos para a operação da máquina, de forma que um programa escrito pelo usuário final possa simplesmente explicitar uma instrução do tipo “acesse o endereço da internet www.yahoo.com, e requisiite uma página html daquele servidor. Leia os dados da página html fornecida”, sem se preocupar com nenhum dos literalmente centenas de detalhes intermediários relativos à conexão existente com a internet.

Ou seja, não são todas as pessoas que desejam programar um computador que precisam compreender e consultar os padrões estabelecidos pelos fabricantes do computador ou do microprocessador. Na maior parte das vezes só se precisa conhecer os recursos disponibilizados ao programador pelo Sistema Operacional, ou as vezes nem isso.

Quando se decide fazer um programa de computador, deve optar-se por uma linguagem a ser utilizada. Como para comunicação entre pessoas temos português, francês, inglês e outras línguas, há uma série de padrões especificados que compõem diversas linguagens nas quais se pode escrever um programa.

Uma classificação geralmente usada para essas linguagens de programação divide-as em “linguagens de alto nível”, e “linguagens de baixo nível”. Sem uma referência absoluta, pode-se dizer que uma dada linguagem é de mais alto nível que uma outra quando a primeira se parecer mais com um idioma escrito e disponibilizar uma representação mais abstrata do computador, que permita o programador precise se preocupar menos com detalhes relativos ao sistema operacional e a parte física da máquina.

Assim sendo, a linguagem disponibilizada pelo fabricante do microprocessador, é a de mais baixo nível possível, a chamada “linguagem de máquina”. Onde cada código numérico corresponde a uma operação lógica. Esta linguagem não é utilizada sequer pelo próprio fabricante. De nível imediatamente acima dessa, o chamado “assembler”, é uma em que cada instrução lógica ganha um nome. Mesmo assim são escritas uma a uma. Um arquivo assembler é escrito como se fosse um arquivo texto, e em seguida transformado para um conjunto de dados que pode ser interpretado diretamente pelo microprocessador por um programa chamado “montador” ou as-

sembler em inglês. Por razões históricas, em português do Brasil a linguagem em si é chamada de assembler, como em inglês.

Acima do assembler há todo um leque de linguagens que já passam a não depender mais do microprocessador que se usa, mas, dependendo-se da finalidade do programa, pode ou não depender do sistema operacional, do computador ou mesmo de um microprocessador específicos.

Uma das linguagens mais utilizadas é o C. Considerada de “nível médio”, grande parte do código dos próprios sistemas operacionais é escrita na mesma - sendo que alguns pontos críticos tem que necessariamente ser escritos em assembler. Mas trata-se também de uma linguagem genérica, e bem eficiente pelo fato de que por ser de nível baixo, cada instrução em C corresponder a apenas uma, ou a um conjunto pequeno de instruções lógicas em código de máquina. Ao se completar um programa em C, novamente um arquivo texto legível em qualquer editor de textos, ele é processado por um programa denominado “compilador”, que como o “montador”, faz uma tradução do programa para uma sequência de dados que possa ser interpretada diretamente pelo microprocessador.

Como o C as outras linguagens também são compiladas ou interpretadas por programas intermediários. Um programa interpretado, em oposição a um compilado, não é traduzido diretamente para uma sequência de dados que o microprocessador processe diretamente, e sim, lido passo a passo, na sua forma de texto mesmo, por um outro aplicativo, que então executa para cada instrução do código a ser interpretado, as tarefas equivalentes no nível do microprocessador ou do sistema operacional. As linguagens de script, embutidas em aplicativos, são via de regra, interpretadas.

Uma outra linguagem, muito menos usada, mas interessante justamente no contexto em que as estudamos, é o javascript. Como o próprio nome diz, trata-se de uma linguagem de script, e de fato, está embutida numa classe de aplicativos. No caso, trata-se dos navegadores para a Web. O Javascript se apresenta como uma linguagem intimamente associada com o HTML, e podemos destaca-lo também por ser uma das únicas, senão a única, linguagem de programação que acompanha o sistema operacional Windows da microsoft, em contraste com outros sistemas que podem instalar opcionalmente dezenas de linguagens de programação diferentes, tanto para permanecerem a disposição do usuário, como para controlar partes do próprio Sistema, que ficam então a disposição do usuário para que ele possa ajusta-las as suas preferências se o desejar.

Mas sendo a escrita para a Web justamente uma habilidade interessante para se ter no currículo, então por que não a programação justamente com o javascript?

O que exatamente o javascript faz ou pode automatizar? Uma página Web que utilize só HTML, como foi visto no capítulo 3, tem um conteúdo estático. O que significa que tem um conjunto de textos, e comandos de marca para formatar este texto, mas uma vez no computador em que a página será lida, este conteúdo não se altera. Ele pode ser rolado para cima e para baixo, podem haver formulários que podem ser preenchidos. Mas se a página for apenas em html, nenhuma ação será tomada. Qualquer que seja a executada ação sobre os formulário, o conteúdo da página não é mais que texto com figuras, e não será alterado.

O javascript no entanto, permite que pequenos programas sejam embutidos na página, mesclados ao próprio código HTML. Por vezes, nem um programa inteiro, apenas um ou dois comandos de javascript embutidos no código HTML, já podem

fazer com que uma página deixe de ser estática.

5.4 Dois Exemplos de Programa

Dependendo do que se deseja fazer, uma página HTML assim pode funcionar como um curto programa para uso pessoal, que pode tanto ser colocado na internet para ser usado por todas as pessoas, como ser usado apenas como um computador local.

A figura 5.3 mostra o código html que com apenas 3 expressões em javascript funciona como uma prática calculadora (figura 5.4) que dispensa que a cada operação se digite o operador ou o sinal de igual:

```

<html >
<body >
Calculadora:
<form action="exemplo.html" method="post" >
<input type="text" name="op1" size="4"
onkeyup="this.form.result.value=eval
(this.value + this.form.operacao.value + this.form.op2.value)"
<
<select name="operacao"
onchange="this.form.result.value=eval
(this.form.op1.value + this.value + this.form.op2.value)"
<

```

```

i option;+i /option;
i option;-i /option;
i option;*i /option;
i option;/i /option;
i /select;


```

Os comandos, tanto de HTML quanto de Javascript acima podem parecer intimidadores a primeira vista. No entanto, basta ter uma referência com exemplos das duas linguagens a mão para que o mesmo se reduza à sua simplicidade. E, de fato, a mais vasta biblioteca de exemplos tanto de Javascript quanto de HTML é a própria Internet. Não só pode se olhar como foi feita cada página que visitamos, como existem online milhares de exemplos, e referências completas das duas linguagens.

Assim, pode-se estudar outros usos para pequenos programas feitos em javascript, de forma que sejam simultaneamente abordados problemas tanto de uma

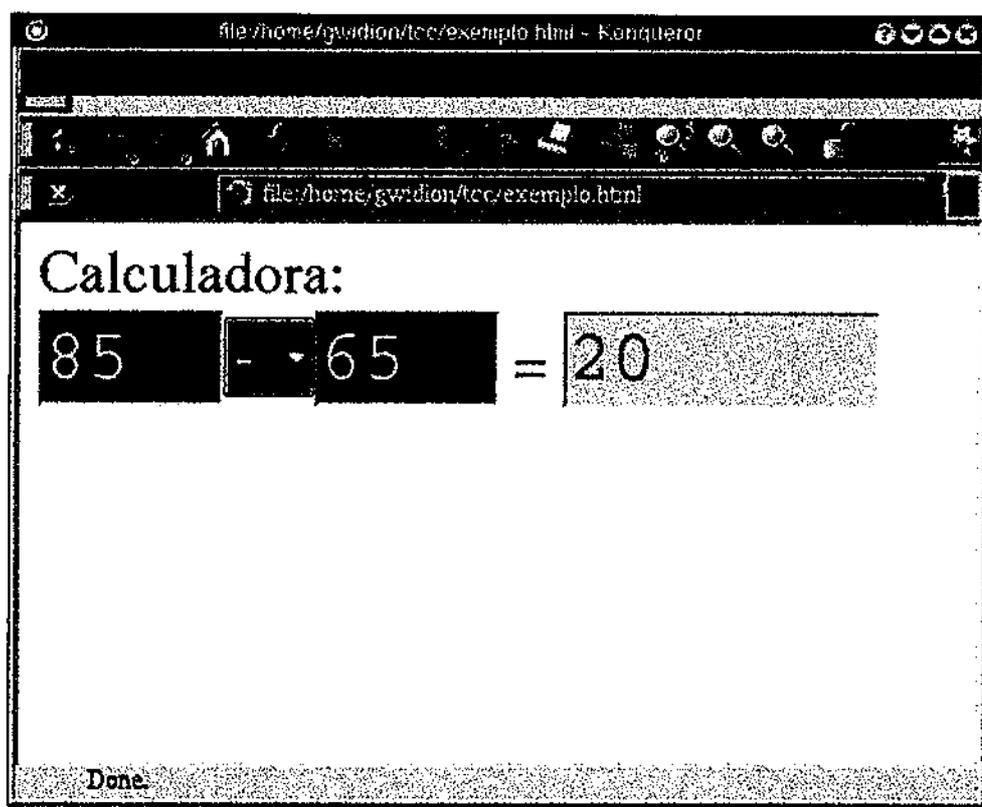


Figura 6: A Calculadora

disciplina qualquer, quanto da programação em si. De fato, no capítulo anterior foi sugerido que a física em particular poderia se beneficiar por uma série de simulações razoavelmente simples de serem programadas, ao mesmo tempo que extremamente ilustrativas.

Sendo assim, pode-se apresentar aqui o exemplo de uma simulação de cinemática - diga-se um corpo atirado para cima e para a frente, com uma dada velocidade inicial. Esse corpo descreve uma curva e cai de volta. A trajetória desse corpo pode ser definida pelas equações:

$$x = V_{0x}t$$

$$y = V_{0y}t - \frac{1}{2}gt^2,$$

Onde t é o tempo transcorrido desde o arremesso, V_{0x} e V_{0y} as componentes horizontal e vertical da velocidade inicial, respectivamente, as componentes horizontal e vertical da velocidade inicial, respectivamente, e g a aceleração gravitacional.

Pode-se fazer um programa que simplesmente desenhe a trajetória do corpo baseado nessas equações. Mas isso não seria muito instrutivo do ponto de vista físico. Por outro lado, pode-se fazer um simulador que desenhe o corpo em sua posição atual, e a cada dado intervalo de tempo atualize a posição do corpo de acordo com a sua velocidade atual, e ao mesmo tempo atualize a sua velocidade de acordo com a aceleração sofrida neste intervalo de tempo. Isso é a simulação do sistema físico.

Posteriormente, ao modificar os parâmetros e experimentar com modificações do programa, assistindo várias vezes a trajetória do corpo animado, o aluno vem a sentir a relação que a fórmula anterior tem com o sistema, não apenas entendendo o significado matemático da fórmula, mas percebendo a relação entre a ação contínua, e a que é possível medir, ou simular, utilizando-se de artifícios tais como intervalos pequenos de tempo, ao invés de um acompanhamento contínuo do corpo.

Um pequeno programa que faz essa simulação poderia ser:

```
html
head
script language="javascript"
var x, y, vx, vy, g;
```

```

var simulacao, ligado=false;
var escala=200/1; //50 pixel/1 metro
var escala"tempo=1/20;
var chao=390;
var dt=100;

function iniciar()
-
corpo=document.getElementById("corpo");
corpo.style.top=chao+"px";
corpo.style.left="20px";
x=20;
y=chao;
g=parseFloat(document.forms[0].g.value);
v0=parseFloat(document.forms[0].v0.value);
alpha=parseFloat(document.forms[0].alpha.value)/180 *Math.PI;
vx=Math.cos(alpha)*v0;
vy=Math.sin(alpha)*v0;
if (ligado)
-clearTimeout(simulacao); "
simulacao=setTimeout("simular()", dt) ;
//executa a animacao dentro de dt ms
ligado=true;
document.forms[0].pausa.value="Pausa";
"

function pausar()

```

```

-if (ligado)
-ligado=false;
clearTimeout(simulacao);
document.forms[0].pausa.value="Continua";
"

else
-ligado=true;
setTimeout("simular()", dt);
document.forms[0].pausa.value="Pausa";
"
"

function simular()
-
corpo=document.getElementById("corpo");
x+=(vx*escala*escala*tempo)/(1000/dt);
y-=(vy*escala*escala*tempo)/(1000/dt);
vy-=g;

if (y<(chao) )
-y=chao;
vy=0;
vx=0;
"

corpo.style.top=y+"px";
corpo.style.left=x+"px";
simulacao=setTimeout("simular()", dt);

```

```

"
i/script¿
i/head¿
i/body¿
i/form action="#" method="post"
style="border: 2px black solid; width: 300px"¿
Velocidade inicial =input type="text" size="4"
value=40 name="v0"¿
m/s i/br¿
Ângulo do arremesso:input type="text" size="4"
value=30 name="alpha"¿
i/sup¿o i/sup¿i/br¿
Gravidade =input type="text" size="4" value="9.8" name="g"¿
m/ls-2 i/br¿
i/input type="button" value="Iniciar" onclick="iniciar()"¿

i/input type="button" value="Pausa" name="pausa"
onclick="pausar()"¿
i/form¿
Escala: 200px =1m i/br¿
tempo da simulacao =tempo real /20 i/br¿

i/div id="chao" style="position: absolute;
left: 0px; top: 400px; width: 100%"¿
i/hr style="border: solid"¿
i/div¿

```

```
i div id="corpo" style="position: absolute;  
left: 20px; top: 390px; width: 1em; *i*/div;
```

```
i/body;
```

```
i/html;
```

Novamente, o código que poderia parecer intimidador não é na verdade complicado. Trata-se claro, por mais simples que seja, de um programa interativo, utilizando uma interface gráfica.

Assim sendo, a maior parte do código acima trata apenas da apresentação do layout dos controles, e do processamento da entrada. A simulação propriamente dita, os cálculos dos parâmetros físicos do corpo, é feita em apenas três linhas:

```
x += (vx * escala * escala * tempo) / (1000/dt);  
y -= (vy * escala * escala * tempo) / (1000/dt);  
vy -= g;
```

Nas três linhas acima, executadas a cada 100 milissegundos, faz-se a imitação, no mundo virtual, do fenômeno físico real. E mesmo nessas três linhas, quase todas as operações são dedicadas a operação de escala, de forma que os efeitos possam ser melhor visualizados. Ou seja, na simulação, um corpo arremessado um metro para cima, não é mostrado subindo um metro em relação ao monitor, e sim, alguns centímetros. E se esse fenômeno ocorreria em .3 segundos, ele é também mostrado em “câmera lenta”, graças a escala de tempo embutida no programa.

O entendimento do funcionamento desse programa, quando na escola se explica sobre velocidade e aceleração, ou, num estágio mais avançado, sobre derivadas e integrais, oferece um ângulo de visão substancialmente inovador em relação a maneira como esses conceitos são normalmente ensinados. O abstrato “diferencial de tempo” é visto em operação, como um “delta t” pequeno o suficiente para não interferir nos resultados visuais. Mesmo essa relação entre diferencial de tempo e intervalo de tempo pode ser entendida de uma forma diferente.

É importante observar que, caso venha a prevalecer no mundo a idéia de computador como caixa preta, tais exemplos nunca poderiam vir à luz num ambiente escolar. A idéia de relegar a programação do computador a círculos estritos de poucas empresas autorizadas a fazer programas, e deixar aos usuários a opção de escolher que site ver, ou a qual canal assistir, privando-os de todas as outras funcionalidades hoje existentes.

6 Conclusão

Do que foi exposto, restam alguns pontos sobre os quais não recaem dúvidas: a informática trouxe a civilização a um ponto de transição, um vértice. Algumas áreas do conhecimento podem já ter assimilado essas mudanças, enquanto que outras ou não se aperceberam de todo o potencial de mudanças, ou deliberadamente esperam que se faça a mudança por elas.

Mesmo sem que se tenha desenvolvido trabalho de campo, não é preciso ir longe para se aperceber que a educação se encontra no segundo grupo.

Se é verdade que escolas particulares de grife todas tem “informática” em destaque em seus folders e outdoors, também é verdade que nessas escolas o que se tem é um período isolado de aulas de informática, com o uso de softwares fechados e proprietários - que não propiciam a criação do aluno, e sim a exposição a um conteúdo multimídia pré elaborado, em que as escolhas por vezes se limitam a qual vídeo assistir primeiro - nada inovando sobre, por exemplo, um livro de papel onde se pode escolher a qual capítulo ler primeiro.

Paralelamente, as mesmas escolas não fazem também distinção entre software livre e software proprietário, apresentando um computador carregado com software

proprietário como algo natural, e nada mencionando sobre o seu funcionamento.

Os estabelecimentos públicos de ensino, por outro lado, por vezes chegam a ser dotados de laboratórios de informática que não são sequer usados, permanecendo trancados por meses a fio justamente por incompatibilidade curricular, que inclui a falta de pessoal capacitado para a operação das máquinas numa atividade de ensino - mesmo que seja como as dos estabelecimentos particulares.

Em todos esses cenários, nada é feito para se contrapor a tendência de transformar os computadores numa TV complicada, com resquícios de máquina de escrever.

Conforme já comentado, tampouco seria uma solução a inclusão de uma ou duas horas semanais no laboratório de informática, numa atividade que roubasse o tempo, por vezes já escasso, de outras disciplinas curriculares. A saída para uma modificação curricular gradual, no sentido da que se aponta necessária neste trabalho, passa necessariamente pela tomada de consciência dos Educadores em si do potencial inerente a informática hoje, juntamente com a consciência dos cenários que se vislumbram caso se negligencie essa área durante a própria formação dos cidadãos.

Essa tomada de consciência por sua vez, não pode se dar por outra forma que não seja a de eventos públicos que discutam os aspectos expostos neste trabalho, aos quais os educadores sejam incentivados, mas não obrigados, a comparecer.

Lembrando que são aspectos que não interessam à “mão invisível do mercado” -portando, não adianta esperar que os estabelecimentos particulares assumam novamente a dianteira, nesse currículo integrando a informática. A dianteira que lhes

cabia, de levar um espaço onde existem computadores até a escola, já foi tomada, e o ensino público já lhe corre atrás nesse aspecto.

Verificado pelos educadores de cada instituição de ensino a necessidade - ou não - de se integrar a informática ao ensino das diversas disciplinas, os estabelecimentos então devem tratar com pessoas habilitadas - com conhecimentos tanto da área técnica quanto de aspectos educacionais - que possam, junto à cada estabelecimento, desenvolver um trabalho de preparo do corpo docente, bem como supervisionar a configuração dos computadores e o software instalado.

Por sua vez, de onde viriam essas pessoas? Desta feita um ponto em que a “mão invisível” pode atuar. Em se existindo demanda das escolas por pessoas com essas capacidades, ou seja, “demanda de mercado”, as pessoas capazes costumam aparecer, quer individualmente, quer como parte de empresas.

Quanto ao software livre, é importante enfatizar não só a questão do custo - visto que há uma significativa economia de recursos financeiros, uma vez que softwares proprietários teriam que ser licenciados individualmente para cada máquina dos laboratórios - como também a questão de interoperabilidade: um software livre necessariamente tem documentada a forma que seus documentos são formatados, permitindo que se escrevam outros programas que leiam os mesmos documentos. Desta forma se mantém sempre a liberdade de trocar um programa por outro, sem a perda do conteúdo já gerado no programa antigo.

E, além do custo e da interoperabilidade, o aspecto mais importante, de se poder saber o que cada programa faz, e interferir no seu próprio funcionamento, adequando-o as necessidades específicas de cada grupo de usuários.

Com essas considerações, com os exemplos vistos nos capítulos 4 e 5 de situações possíveis de ensino dentro desta proposta, e sem deixar que se perca de vista a importância de uma ação social nessa área, ao menos no que tange as relações com a escrita, encerra-se este trabalho. ,

Referências Bibliográficas

- [1] PLATÃO. Platão: Vida, obra, doutrina. In: PLATÃO (Ed.). *Diálogos I: Mênon, Banquete, Fedro / Platão*. 21. ed. Rio de Janeiro: Ediouro, 1999.
- [2] RIPPER, A. V. O preparo do professor para as novas tecnologias. In: OLIVEIRA, V. B. d. (Ed.). *Informática em Psicopedagogia*. 1. ed. Campinas: Editora Senac, 1997.
- [3] ALVES, M. J. P. *Construindo Supercomputadores Com Linux*. Rio de Janeiro: Brasport, 2002.
- [4] CASTRO, C. d. M. *Educação na Era da Informação*. 1. ed. São Paulo: Universidade, 2001.

