
BOCA-LAB: Corretor automático de Código adaptado ao Ensino de Linguagem de Programação.

Débora Cabral Nazário¹, Allan de Souza¹

¹Departamento de Ciência da Computação – DCC
Universidade do Estado de Santa Catarina (UDESC)
Campus Universitário Prof. Avelino Marcante s/n - Bairro Bom Retiro -
Joinville – SC
Brasil CEP 89223-100

debora@joinville.udesc.br, lupercia@gmail.com

Abstract *The computational programming subject is usually the first contact of students with the computational logic. The objective of this work is to reduce the difficulties encountered by the student, in that subject, adapting the correction system BOCA, which is used in programming competition. The tool proposed, in addition to the automatic correction, performs the orientation by content to the submitted problem by the student, through tips. The results has shown that the system improves the process of learning programming language.*

Resumo. *A disciplina de programação é geralmente o primeiro contato do estudante com a lógica computacional. O objetivo deste trabalho é reduzir as dificuldades encontradas pelos estudantes, nesta disciplina, adaptando o sistema de correção BOCA, que é utilizado nas maratonas de programação. A ferramenta proposta, além da correção automática, realiza a orientação por conteúdo a cada problema submetido pelo estudante, através de dicas. Pelos resultados obtidos, verifica-se que o sistema contribui no processo de aprendizagem de Linguagem de Programação.*

1. Introdução

A construção do conhecimento, de soluções criativas e de consciência crítica é favorecida na sociedade moderna pelo avanço tecnológico, que proporciona cada vez mais a necessidade do aprendizado contínuo [Silva 2000]. Dessa forma, pode-se afirmar que a tecnologia contribui nos métodos de aprendizagem, na medida em que possibilita o acesso rápido ao conhecimento, proporcionando uma maneira mais atraente e diferente das provenientes dos métodos clássicos de ensino. De acordo com [Gottschalk 2004] toda tecnologia para se julgar apropriada deve passar por um rigoroso planejamento e entendimento da necessidade do aluno e do curso.

A disciplina de linguagem de programação está presente em vários cursos de computação, na grade curricular, normalmente é apresentada na primeira fase do curso e tem como objetivo permitir com que o aluno desenvolva a lógica de programação, o tornando capaz de construir algoritmos corretos e otimizados [Júnior 2004].

Um corretor automático de código pode ser utilizado para facilitar o processo de exercício e prática, fundamental na abordagem atualmente utilizada no ensino de linguagem de programação. A contribuição dos corretores está ligada na comodidade, facilidade e agilidade que pode proporcionar ao estudante.

O objetivo principal deste trabalho é produzir um sistema web que possa receber submissões dos estudantes com seus respectivos exercícios e corrigi-los automaticamente. Isto permite a redução do tempo de resposta, evita a quebra de raciocínio lógico, diminui a carga de correção sobre o professor, além de procurar motivar o aluno para um melhor aproveitamento na disciplina.

2. Fundamentação Teórica

Para [Júnior 2004] "A questão chave no processo é justamente como motivar o aluno, fazendo-o tomar gosto pelo aprendizado, procurando superar suas dificuldades, como a falta de habilidades matemáticas", portando o processo de aprendizagem nas disciplinas de programação está diretamente ligado a motivação do aluno. A disciplina de programação possui um dos maiores índices de reprovação nas instituições brasileiras, encontrar uma maneira de tornar a qualidade do processo mais eficaz é uma necessidade atual das abordagens utilizadas [Rodrigues 2004]. O objetivo deste trabalho é proporcionar, através de uma ferramenta de auxílio, uma maneira de facilitar o processo de aprendizagem em linguagem de programação, tornando-o mais atraente e dinâmico.

A aplicação tecnológica nos métodos tradicionais de ensino assume uma abordagem instrucionista, que implementada no computador, o mesmo deverá possuir uma grande quantidade de informações, as quais devem ser passadas para o aluno em forma de tutorial, atividades dirigidas ou exercício-e-prática [Valente 1993]. Este cenário descreve o computador como ferramenta facilitadora do processo de aprendizagem, permitindo com que cada aluno possa ter sua capacidade avaliada individualmente, facilitando o planejamento e o controle das avaliações.

O sistema proposto está apoiado na abordagem instrucionista, utilizando-se das principais características deste modelo: professor como centro do processo, objetivos bem definidos, respostas previamente programadas, uso de exercício-e-prática, mídias de texto e ilustrações em forma de tutoriais, avaliação quantitativa.

A principal tecnologia utilizada no trabalho proposto é a Internet, pois, permite a disponibilidade de lugar ou hora, é de baixo custo operacional, tem grande aceitação e domínio dos alunos. O sistema ficou caracterizado como uma ferramenta de apoio ao ensino presencial atual. O foco em um corretor exclusivo para educação é o diferencial da proposta, permitindo ao aluno uma maior integração com a turma, estimulação ao auto-aprendizado e reforçando a atividade de exercício e prática.

3. Corretores Automáticos de Código

Os corretores automáticos de código possuem como objetivo validar a consistência de um algoritmo através de vários casos teste. Este processo é semelhante ao utilizado pela engenharia de software no teste da caixa preta ou funcional. No teste da caixa preta não se tem conhecimento interno do algoritmo, os dados de entrada são fornecidos, a partir deles o algoritmo interpreta os dados recebidos e o resultado de saída gerado pelo algoritmo é comparado com o conhecido.[Pressman 1995]

Corretores como [Pc2 2008] e [Mooshak 2008] foram estudados, porém o BOCA (*Online Contest Administrator*) foi escolhido por atender melhor as necessidades da proposta, como: possui total integração na web; utiliza a linguagem de programação PHP, que permite adaptações no sistema com facilidade; possui código aberto sem limitação de uso; além de suporte e colaboração do autor Cassio Polpo de Campos, atualmente professor da USP. O corretor automático proposto será composto por partes de programação já existentes no BOCA, formando assim um *plugin*. A principal diferença da proposta, chamada BOCA-LAB, em relação aos atuais corretores de código está na eliminação do ator juiz, na possibilidade do aluno obter dicas de orientação, a medida em que realiza submissões inválidas e o direcionamento do foco da ferramenta para o auxílio da educação e não mais nas competições de programação, como os atuais corretores de código estudados.

4. Testes e Resultados

Com a finalidade de testar a ferramenta desenvolvida foi elaborado para o semestre de 2009/1 um teste de campo prático nas turmas de Linguagem de Programação I, no curso de Bacharelado em Ciência da Computação da UDESC. Esta seção tem como objetivo descrever o cenário no qual foi elaborado o teste de validação da ferramenta, bem como analisar a contribuição da mesma no processo de ensino-aprendizagem.

4.1. Perfil da turma

Os testes foram realizados nas turmas A, B e C. As turmas A e B são turmas com estudantes ingressantes no curso, possuem como característica o mesmo professor, porém com as aulas laboratoriais separadas em horários distintos. A turma C, por sua vez, é turma exclusiva de alunos repetentes na disciplina. A apresentação da ferramenta para ambas as turmas, foi realizada em uma aula expositiva, onde foram demonstradas passo a passo todas as etapas necessárias para o estudante conseguir fazer uso da ferramenta.

Os estudantes receberam como *login* de acesso a sua matrícula, todos os alunos indiferentemente da turma na qual estavam alocados não foram separados no sistema, todos os problemas, atividades e dicas foram concebidas a todos os alunos de maneira igualitária. A utilização da ferramenta foi facultativa ao aluno, não foi aplicado nenhum tipo de penalidade ou perdas para os alunos que não utilizaram o sistema.

4.2. Contextualização dos Exercícios

Para a realização dos exercícios foram adaptados os atuais exercícios aplicados nas aulas presenciais. Desta maneira, procurou-se manter o nivelamento equivalente ao cobrado em sala de aula, em relação à ferramenta. Para possibilitar uma melhor compreensão de cada problema, o mesmo foi contextualizado dentro de um problema / história, procurando sempre fazer ligação da utilização do algoritmo envolvido com o mundo real, desta maneira os exercícios passaram a ter identidade.

Durante os testes foram disponibilizados um total de nove problemas, cada problema foi elaborado de forma a especificar e abranger um determinado conteúdo específico da disciplina.

4.3. Taxa de Utilização do Sistema

A turma AB possui a quantidade de quarenta alunos, a turma C, onze alunos. Para análise dos dados foi considerado como uso válido, o aluno que utilizou o sistema no mínimo uma vez e realizou as atividades de *login* e visualização de problema. A utilização pela turma AB foi de 100% dos alunos e da turma C de 73% (3 alunos não utilizaram o sistema).

Uma outra métrica utilizada para avaliar a ferramenta é a quantidade de estudantes que resolveram no mínimo um problema, ou seja, conseguiram completar todo o ciclo proposto: visualizar e interpretar o problema proposto; desenvolver uma solução em código fonte; submeter para correção automática e receber a avaliação como correta pelo sistema. Na turma AB 92% dos alunos executaram um ciclo completo e na turma C 87%.

Para finalizar a análise da utilização da ferramenta foi calculada a quantidade e a média das correções realizadas pelo sistema. Cada correção corresponde a uma submissão do aluno para um determinado problema. O BOCA-LAB obteve um total de 842 correções, com média de 17,54 correções por aluno. A ferramenta foi disponibilizada para os estudantes em um período de três semanas. Para realizar a correção desse mesmo total de exercício, se cada professor gastar um tempo médio de três minutos por correção, o mesmo gastaria duas horas e quarenta minutos do seu tempo diário para as correções.

4.4. Aproveitamento do Sistema

Uma maneira de poder quantificar a influência que o uso da ferramenta tem sobre a nota do estudante é utilizando o cálculo de correlação. Para [Larson e Farber 2006] a correlação é uma relação entre duas variáveis representadas por um par ordenado. Neste contexto, a formação do par ordenado é constituída da relação: número de problemas solucionados no BOCA-LAB e a nota da prova das aulas presenciais.

O índice de correlação da turma AB foi de 0.4050, o da turma C de 0.4138. Este valor é classificado como uma influência forte entre número de problemas resolvidos e a nota da avaliação, sendo assim uma contribuição de $\approx 40\%$ na probabilidade de um bom desempenho nas avaliações é atribuída ao BOCA-LAB.

Outra maneira de avaliar o sistema proposto foi através da aplicação de um questionário de satisfação respondido por 81% dos estudantes envolvidos. Neste questionário foi priorizada a avaliação do aluno quanto a funcionalidade da ferramenta. As principais citações com relação às vantagens da ferramenta foram ligadas à facilidade de uso, a disponibilidade e a contribuição no ensino. Quanto a avaliação crítica, os alunos reclamaram do número de dicas insuficientes ou não condizentes com a necessidade do estudante, pequena quantidade de problema e principalmente na dificuldade de sincronizar a formatação do programa com a exigida pela ferramenta.

5. Conclusão

Pode-se concluir que a ferramenta obteve sucesso quanto as suas disposições iniciais. O número de estudantes que obtiveram sucesso no uso das atividades propostas foi superior a 90%, este dado comprova o sucesso da ferramenta perante sua aceitação.

Ao significativo número de submissões erradas podem ser atribuídos dois principais fatores. O primeiro fator é a necessidade exata da formatação da saída do programa com as especificações do problema, esse processo pode levar algum tempo para adaptação do aluno, porém ao longo do tempo este fator se mostra de pequena relevância. O segundo fator a se considerar é a característica dos alunos, muitos iniciantes em programação e sujeitos aos diversos erros comuns de lógica e programação.

O direcionamento visando a utilização de corretores automáticos de código exclusivamente para fins educacionais é uma inovação deste trabalho, pois não tem-se conhecimento de projetos com objetivos semelhantes, este fato limita as possíveis comparações entre o sistema proposto e os atuais corretores de código.

Porém o interesse e o reconhecimento da importância das competições de programação não estão descartados, o trabalho faz reuso das interfaces e mantém semelhante à acessibilidade do sistema, possibilitando um aproveitamento do conhecimento adquirido na ferramenta proposta para futuras participações em competições.

Essas atividades de maneira automática podem ser estendidas a outras disciplinas do curso como: banco de dados, algoritmos e estrutura de dados, compiladores, etc. A vantagem deste tipo de sistema, que apesar de inicialmente apresentar um significativo trabalho, ao longo do tempo todo material pode ser reaproveitado, o tornando de fácil uso e baixa manutenção.

Referências

- Gottschalk, T. H. (2004). Guide 1 - Distance Education: An Overview. University of Idaho Engineering Outreach. Disponível em: <<http://www.uidaho.edu/eo/dist1.html>>.
- Júnior, C. E. R. J. C. R. P. (2004). O processo de ensino-aprendizagem de fundamentos de programação: Uma visão crítica da pesquisa no Brasil. WEI - Workshop sobre Educação em Computação, p. 19–21.
- Larson, R. e Farber, B. (2006). Estatística Aplicada. 2a. ed: Pearson.
- Mooshak. (2008). System for Managing Programming Contests. Disponível em: <<http://mooshak.dcc.fc.up.pt/>>.
- PC2. (2008). Programming Contest Control System. Disponível em: <<http://www.ecs.csus.edu/pc2/>>.
- Pressman, R. S. (1995). Engenharia de Software. 3. ed. São Paulo: Makron Book do Brasil.
- Rodrigues, M. C. (2004). Como ensinar programação? Informática - Boletim Informativo Ano I n_01. ULBRA. Canoas, RS, Brasil.
- Silva, R. W. A. (2000). Educação a Distância em Ambientes de Aprendizagem. Matemática Auxiliada pela Realidade Virtual. UFSC. 2000. Disponível em: <<http://teses:eps.ufsc.br/defesa/pdf/7882:pdf>>.