

O Desafio de Trabalhar com Alunos Repetentes na Disciplina de Algoritmos e Programação

Janice Inês Deters¹, Júlia Marques Carvalho da Silva²,
Elisângela Maschio de Miranda², Anita Maria da Rocha Fernandes³

¹Serviço Nacional de Aprendizagem Industrial (SENAI)
Rodovia SC 401, 3730 - Saco Grande - 88032-005 - Florianópolis - SC

²Laboratório de Inteligência Aplicada - Universidade do Vale do Itajaí
Rua Uruguai, 458 – 88.302-202 – Itajaí – SC – Brasil

³Laboratório de Inteligência Aplicada - Universidade do Vale do Itajaí
Rodovia SC 407 – KM 4, Sertão do Maruim – 88.122-000 – São José – SC – Brasil
jancd_2000@yahoo.com.br, julia@univali.br, elis@univali.br,
anita.fernandes@univali.br

Resumo. Este artigo descreve a problemática da repetência nas disciplinas voltadas ao ensino de algoritmos e de programação e o seu impacto na vida do aluno, bem como as dificuldades encontradas pelos professores em trabalhar com alunos repetentes. São apresentadas algumas ferramentas computacionais de apoio ao ensino e aprendizagem, direcionadas ao aprimoramento da lógica de programação do aluno, utilizadas pelas autoras do artigo em diferentes Instituições de Ensino Superior (IES), bem como os resultados alcançados com a sua utilização.

Palavras-chave. Ensino de Programação. Ferramentas Educacionais. Alunos repetentes.

Abstract. This paper describes the problem of repeater in the disciplines introductory programming and its impact in the student's life, as well as the difficulties found by teachers in working with repeater students. Some computational tools are introduced to support the learning, aiming to improve the programming logic of the student, used by the authors of the paper in different Universities, as well as the results reached in its use.

Keywords. Introductory programming teaching. Educational tools. Repeated students.

1. Introdução

As disciplinas que envolvem o ensino de Algoritmos e Programação são normalmente lecionadas nas primeiras fases dos cursos das áreas tecnológicas. Tais disciplinas são consideradas desafiadoras pelos alunos, pois exigem o desenvolvimento de estratégias de solução de problemas com base lógico-matemática. A consequência disso é o elevado

número de problemas de aprendizagem, favorecendo a ocorrência de reprovações e desistências.

O conteúdo programático geral destas disciplinas aborda temas como Conceitos Fundamentais, Representação de Dados, Operações, Instruções Primitivas, Desvio Condicional, Laços de Repetição, Tipos Compostos e Modularização; e a carga horária varia de 60 a 120 horas/aula (quando a prática da programação é exigida em conjunto). Ainda, dependendo da fase em que a disciplina é lecionada, o ensino pode estar atrelado à utilização de alguma linguagem de programação para a construção de programas. Entretanto, não existe um padrão definido, cabendo a cada IES selecionar e organizar a estrutura de seu curso conforme a necessidade de ensino ou de mercado.

Este artigo surgiu a partir do desafio em trabalhar com alunos repetentes nas disciplinas de Algoritmos e Programação. Tais dificuldades foram compartilhadas entre docentes de IES diferenciadas, gerando encontros para a discussão dos problemas encontrados e como enfrentá-los. Para cada cenário são utilizadas estratégias de ensino e de interação diversas, envolvendo o uso de recursos tecnológicos para a realização das atividades práticas e distribuição dos materiais didáticos. Entretanto, percebe-se a necessidade de motivar o aluno repetente, sendo desenvolvidas algumas estratégias para estimular o seu aprendizado e a sua consciência como responsável pelo aprendizado, convertendo uma situação de fracasso em uma situação de sucesso.

Ao longo deste artigo será apresentada uma visão geral sobre os problemas de aprendizagem nas disciplinas de algoritmos e programação, o cenário atual da média de reprovações nestas disciplinas e experiências vivenciadas em IES com cursos de graduação, tecnológico e técnico. Também serão narradas as estratégias aplicadas para reverter esse quadro e os resultados alcançados com sua aplicação.

2. Problemas de Aprendizagem

Mayer (1988) afirma que a pesquisa do ensino e aprendizagem de algoritmos e programação pode ser analisada sobre três fases: uma fase inicial, na qual muitas fortes reclamações foram apresentadas alegações quanto os resultados esperados e os melhores métodos de instrução de programação de computador, uma observação em fase de montagem que os dados apontaram para problemas nos alunos a aprender, e uma fase de pesquisa em teoria baseada em estudos que foram realizada sistematicamente para compreender os processos subjacentes à programação de ensino e aprendizagem.

Soloway e Ehrlich (1984) discutem como o acadêmico pode obter as habilidades de um programador, e para isto, é necessário que sejam capazes de estabelecer planos (fragmentos que representam ações sequenciais programadas) e regras de programas (convenções e composição dos planos). Ainda, apresentam atividades a serem aplicadas aos alunos como, por exemplo, completar um trecho de código em um determinado algoritmo.

Resolver problemas inclui uma predisposição para o questionamento de quais os processos cognitivos que devem ser utilizados, a fim de gerar o conhecimento necessário à resolução do problema. Envolve as operações de pensar, analisar, sintetizar e de avaliar (Falckembach & Araújo, 2004).

Menezes e Nobre (2002) apontam três problemas gerais: (i) o elevado número de acadêmicos por turma, que inviabiliza a realização de um acompanhamento individualizado; (ii) avaliações que ocorrem apenas por meio de provas escritas ou trabalhos individuais, não promovendo uma evolução gradual da aprendizagem; (iii) heterogeneidade da turma, disparidade de conhecimento e ritmo de aprendizagem.

Dentre as dificuldades vivenciadas pelos professores de algoritmos em sala de aula, Menezes e Nobre (2002) relacionam: (i) a capacidade de reconhecer habilidades inatas de seus alunos; (ii) a apresentação de técnicas de resolução de problemas; (iii) como promover o desenvolvimento da capacidade de abstração do aluno, permitindo-o selecionar as estruturas de dados coerentes; (iv) como facilitar a cooperação e colaboração entre os alunos.

Rodrigues Jr. (2004) aponta como problema o desconhecimento da importância da disciplina para a formação do acadêmico. Esse fato é comum tanto para alunos sem experiência prévia, pois não tem a dimensão da aplicação dos conteúdos na prática, quanto para alunos que já programam em alguma linguagem, pois não possuem consciência científica e metodológica do trabalho, apresentando apenas conhecimento suficiente para o desenvolvimento de soluções de natureza restrita.

Já sob o ponto de vista dos acadêmicos, Tobar et al. (2001) definem alguns motivos que acarretam a frustração perante a disciplina: (i) a preocupação excessiva com detalhes de sintaxe da linguagem sendo usada; (ii) a falta de uma visão sobre o que se quer solucionar, como idealizar soluções adequadas, e consequentemente mapear essas soluções em passos sequenciais e abstrair o funcionamento dos mecanismos escolhidos; (iii) o estabelecimento de um raciocínio lógico visando à resolução de problemas, com base em um modelo incremental, em relação à complexidade e à estratégia de refinamentos sucessivos.

Raabe e Silva (2005) identificaram três tipos de aspectos que contribuem para a dificuldade de aprendizagem de algoritmos:

- Problemas de natureza didática: são aqueles que envolvem a diversidade de cultura e de experiência dos alunos ingressantes, a excessiva quantidade de alunos por turma, o que inviabiliza um atendimento e avaliação individual, a dificuldade de expressar e compreender a lógica desenvolvida e a ausência de materiais de referência de qualidade aos alunos. Além disso, têm-se problemas particulares, que dizem respeito à escolha equivocada do curso.
- Problemas de natureza cognitiva: consiste em aspectos que normalmente estão atrelados a vivência precedente ao ingresso à universidade. São encontrados em alunos cujos estudos foram interrompidos há algum tempo, ou que o ensino médio não os permitiu desenvolver adequadamente as faculdades cognitivas.
- Problemas de natureza afetiva: um aluno pode estar convicto da sua escolha quanto ao curso, ter aptidão para resolução de problemas que envolvem raciocínio lógico, e mesmo assim não consegue obter o sucesso na disciplina. Nesses casos, o foco deve ser direcionado a possíveis causas afetivas que interferem no desenvolvimento. Goleman (1999) cita que alunos que apresentam sentimentos de ansiedade, raiva ou depressão têm dificuldade em

aprender. Esses estados afetivos criam um bloqueio na mente, não permitindo que a informação passada pelo professor seja captada de forma eficiente. A diversidade de problemas desta natureza é alta, entretanto neste trabalho optou-se por classificar esses pela sua frequência, isto é, ocorrem esporadicamente ou manifestam-se ao longo de toda a disciplina.

2.1. Ferramentas de Apoio a Aprendizagem de Algoritmos e Programação

Quando se trata de problemas de aprendizagem de Algoritmos e Programação, é comum encontrar ferramentas que visam apoiar o conteúdo abordado nas disciplinas. Sobre o ensino de Algoritmos, existem ferramentas para apoio ao desenvolvimento de atividades que estimulem o raciocínio lógico e que simulem a execução de programas utilizando meta-linguagens. Alguns exemplos são:

- o ambiente virtual de ensino-aprendizagem ALICE (Raabe & Silva, 2005): permite ao professor acompanhar o desempenho do acadêmico no decorrer do semestre, através do desenvolvimento pelo acadêmico de exercícios inseridos no ambiente pelo professor. Também é possível inserir a matéria do semestre no ambiente, entre outros.
- o editor WebPortugol (Hostins, 2006): para construção de algoritmos em meta-linguagem Portugol;
- a IDE CodeBlocks (2008): que permite o desenvolvimento de algoritmos nas linguagens C e C++, sendo também utilizado na área de jogos.

Já no ensino de Programação é mais comum o uso de IDEs, isto é, editores próprios para as linguagens de programação que estão sendo ensinadas. Neste caso o aluno não fica preso a uma ferramenta, podendo selecionar o editor que mais se familiariza.

- editor NetBeans (2008): utilizado no ensino da linguagem Java, por oferecer uma fácil infra-estrutura de compilação e execução.
- Bloco de Notas, por exemplo: utilizado no ensino da linguagem PHP. Neste caso, o aluno fica livre para selecionar qualquer editor de texto, enquanto os exercícios são enviados a um servidor web único, facilitando o processo de ensino.

Além dessas ferramentas apontadas, na literatura encontra-se o relato de outras experiências realizadas como uso de ferramentas como o Portugol/Plus, VisualG, BlueJ, etc.

3. Cenário

As disciplinas de “Algoritmos” e “Programação” possuem um dos maiores índices de reprovação nos cursos de tecnologia. Segundo Rocha (2003), as disciplinas de lógica, algoritmos e programação iniciam com uma média de 50 alunos, e em poucos meses constata-se que a taxa de reprovação (ou desistência) chega a 60%. Segundo Galhardo (2006) os altos índices de reprovação estão relacionados diretamente a forma como ocorre o processo de ensino-aprendizagem. No processo de ensino-aprendizagem existem quatro componentes que interagem: aluno, professor, disciplina a ser lecionada e as condições ambientais, que quando relacionadas adequadamente fazem com que o

processo flua sem apresentar problemas. Mas quando algo de errado acontece com um ou mais componentes o processo passa a ser dificultado, podendo culminar em vários problemas, dentre eles o alto índice de reprovação na disciplina em questão.

A Tabela 1 apresenta uma visão geral das disciplinas verificadas neste artigo. Para cada uma são apresentados os índices de reprovação por disciplina (no último semestre vigente), o curso a que estão vinculadas, os recursos utilizados e temas ensinados, e as estratégias de ensino e avaliação.

Tabela 1. Cenários das disciplinas

Nome da Disciplina	Curso	Índice de repetência	Recursos utilizados e temas ensinados	Estratégias de Ensino e Avaliação
[1] Algoritmos e Programação (1º período)	Ciência da Computação	56,41%	Meta-linguagem Portugol Ambiente Virtual ALICE Editor WebPortugol IDE CodeBlocks	Aulas expositivas com exercícios práticos (listas de exercícios), trabalhos em grupo e avaliações individuais
[2] Algoritmos e Programação (2º período)	Ciência da Computação	50%	Ambiente Virtual ALICE IDE CodeBlocks	Aulas expositivas com exercícios práticos (listas de exercícios), trabalhos em grupo e avaliações individuais
[3] Programação 1 (3º período)	Ciência da Computação	20%	Ambiente Virtual genérico Linguagem JAVA NetBeans	Aulas expositivas com exercícios práticos (listas de exercícios) e projeto final
[4] Lógica e Algoritmos (1º período)	Tecnologia em Sistemas para Internet	37,78%	Meta-linguagem Portugol Ambiente Virtual ALICE Editor WebPortugol	Aulas expositivas com exercícios práticos (listas de exercícios) e avaliações individuais
[5] Programação 1 (1º período)	Tecnologia em Sistemas para Internet	40%	Ambiente Virtual genérico Linguagem de Web PHP	Aulas expositivas com exercícios práticos (listas de exercícios) e avaliações individuais
[6] Algoritmos e Programação (1º período)	Curso Técnico de Programação	50%	Ambiente Virtual da IES VisualG	Aulas expositivas com exercícios práticos (listas de exercícios) e avaliações individuais

Nas disciplinas [1, 2 e 4] é incentivada a utilização do ambiente ALICE e do WebPortugol para uso extra-classe, o que nem sempre ocorre por motivos variados, como trabalhar em tempo integral, não ter Internet em casa ou mesmo pela falta de interesse em estudar extra-classe. O acadêmico encontra, no ambiente ALICE, o material teórico referente a disciplina, bem como exercícios a serem respondidos e que são corrigidos pelo professor da disciplina (que insere qual a nota retirada pelo aluno e os problemas encontrados na resolução do exercício). Já o Editor WebPortugol é utilizado para que os acadêmicos possam visualizar a execução de seus programas na metalinguagem Portugol.

Ainda, nas disciplinas [1, 2, 3 e 4] foi possibilitado que o acadêmico enviasse os trabalhos solicitados pela disciplina pelo email, antes do período de entrega. Para os alunos que enviavam antes era corrigido o trabalho e enviado ao aluno os erros encontrados, para que o mesmo pudesse corrigir e reenviar novamente. Note-se que eram enviados os erros, a correção ficava para o acadêmico. A solução foi interessante

para os alunos interessados, notando-se claramente a melhora nos trabalhos a serem realizados e entregues nas médias posteriores. Mas para aqueles alunos que não se interessam em estudar extra-classe não foi de grande valia, pois se interessaram só em enviar uma resposta e mesmo aqueles que enviaram antes não procuraram melhorar os trabalhos.

É necessário ressaltar que, com frequência, turmas apresentam alunos com diversidade social, cultural ou limitações físicas / mentais. No geral, as turmas são formadas por alunos recém-egressos do ensino médio, alunos que não estudam há algum tempo e portadores de necessidades especiais. Recentemente, participaram das turmas alunos com deficiência auditiva, deficiência motora, deficiência mental e autismo.

3.1 Levantamento dos Problemas

Com o objetivo de diminuir os índices de reprovação nas disciplinas de algoritmos e de programação nas IES algumas ações foram tomadas. Num primeiro momento procurou-se identificar os motivos que levaram a reprovação. Nesta etapa foi analisado o comportamento de cada aluno, verificando como e quando ocorreu a reprovação, sendo possível diferenciar os alunos que reprovavam devido à desistência ao longo da disciplina daqueles que reprovavam mesmo cursando toda a disciplina. Tal diferenciação se faz necessária, pois ao cursar novamente a disciplina estes alunos apresentam conhecimento prévio diferenciado. O foco de estudo do presente artigo se encontra nos alunos considerados repetentes, isto é, que permaneceram nas aulas até o final do curso, porém sem obter aprovação.

Percebeu-se que a identificação dos alunos repetentes se trata de uma tarefa complexa, sendo necessário analisar o comportamento do aluno, como responde aos estímulos em sala de aula, sua dedicação e esforço para compreender os temas abordados. Neste sentido, a utilização do ambiente virtual ALICE (Silva & Raabe, 2008) auxiliou, já que seu propósito é apoiar o ensino de algoritmos, com ferramentas e funcionalidades específicas. Ainda, o ambiente encontra-se preparado para auxiliar o professor na identificação do estilo do aluno: sua dedicação, comunicação, busca por materiais, etc. A Figura 1 ilustra o ambiente ALICE.

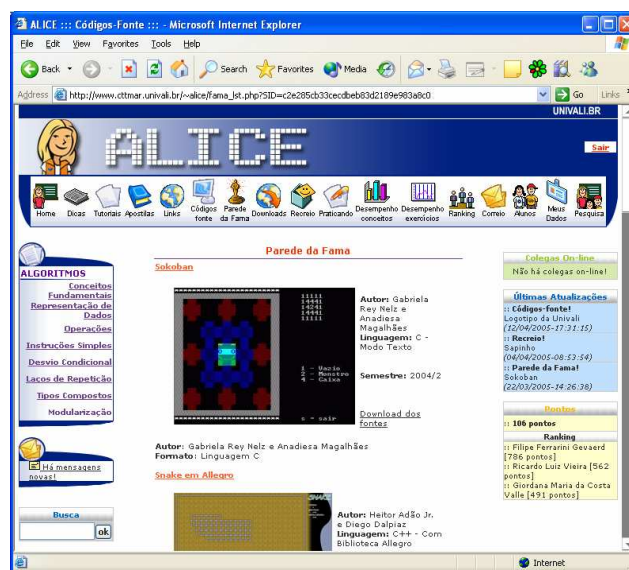


Figura 1. ALICE - Ambiente Virtual para Ensino-Aprendizagem de Algoritmos

Nos demais casos, quando se utiliza ambientes virtuais genéricos (que podem ser aplicados em diversos contextos) e editores de programação isolados, a tarefa do professor em acompanhar, resgatar e analisar os dados de cada aluno se torna maior.

3.2 Medidas Tomadas

A partir desta definição buscou-se compreender os motivos que levaram à reprovação. Os motivos levantados foram categorizados em responsabilidades do professor, aluno e estratégias de ensino. Dentre as ações tomadas para melhorar o processo de ensino e aprendizagem tem-se:

- Mediação pedagógica – foram oferecidos aos professores das IES treinamentos pedagógicos com o intuito de melhorar a didática, a metodologia de trabalho e a forma de avaliação. Além disto, buscou-se auxiliá-lo na elaboração de recursos didáticos e na definição dos objetivos da aula, com o objetivo de tornar a aula mais dinâmica e consequentemente mais interessante ao aluno.
- Incorporação da aprendizagem significativa – segundo Pacheco et al. (2006) a aprendizagem acontece quando ela é significativa. O aluno aprende um novo conceito quando consegue relacionar as suas experiências anteriores e vivências pessoais. Ausubel et al. (1980) define que as três condições necessárias para que a aprendizagem seja significativa são: o material instrucional com os conteúdos organizados de maneira lógica; a existência na estrutura cognitiva do aprendiz do conhecimento organizado e relacionável com o novo conteúdo; e a vontade e disposição do aluno de relacionar o novo conhecimento com aquele já existente.
- Monitoria: os alunos passaram a contar com um monitor em horários extra-classe. O grande número de alunos por sala impede uma aprendizagem mais eficiente, sendo que o professor muitas vezes não consegue dar um acompanhamento individualizado. O monitor tem o papel de auxiliar o professor no processo de aprendizagem e trabalhar as dificuldades individuais do aluno.
- Avaliação – além de avaliar os conteúdos específicos (como era feito até então) são avaliadas também as atitudes necessárias a sua prática profissional.

O segundo passo foi identificar quais os fatores de responsabilidade dos alunos que levam a sua reprovação. Alguns dos fatores identificados são: (i) alunos que trabalham em tempo integral e não possuem tempo para se dedicar aos estudos requeridos na disciplina (cansaço e stress); (ii) pouca dedicação por parte do aluno, que não resolve as atividades práticas propostas e não acessa material de apoio; (iii) dificuldade de aprendizagem e de interpretação de texto - os alunos não conseguem compreender o que o enunciado de uma atividade solicita, levando a uma resolução equivocada; (iv) dificuldade na compreensão do conteúdo proposto, agravada por desconhecerem ou não dominarem conteúdos que são pré-requisitos da disciplina; e (v) alunos que são forçados a estudar algo que não é do interesse, estimulados pela família, mercado de trabalho ou amigos.

3.3 Como Reverter a Situação

Os efeitos da reprovação sobre o aluno são, em sua maioria, negativos. As consequências estão no agravamento do desinteresse e a baixa na auto-estima, fazendo com que se sintam incapazes, perdedores, fracassados, levando-os a deixar o curso. Segundo Novakoski (2008) o impacto da reprovação não se refere somente na disciplina, mas na sociedade, na família, no grupo de amigos.

O aluno reprovado, na sua grande maioria, acredita que a sua reprovação ocorreu devido a uma “injustiça”, o professor o reprovou porquê quis. O aluno não se apercebe no primeiro momento que o insucesso em grande parte é da sua responsabilidade. Sendo assim é necessário desenvolver no aluno a autocrítica, a fim de identificar quais foram os motivos que levaram a repetição, para que a mesma não aconteça novamente. Quando bem conduzido, o aluno aceita a repetência com muito mais naturalidade, e para isso é preciso que ele reflita e reavalie onde aconteceram as falhas.

A reprovação não é uma experiência agradável, porém não é um motivo para desacreditar o estudante, mas sim servir para que ele aprenda o valor do estudo e desenvolva métodos mais eficientes no próximo período. Para que isto aconteça o aluno deve ser levado a mudar a sua atitude no processo de ensino-aprendizagem. Algumas atitudes para tornar isto possível foram tomadas pelas autoras, sendo elas:

- Estimular o aluno a desenvolver as suas habilidades dentro do tempo previsto pela disciplina.
- Fazer com que o aluno se aperceba que ele é responsável pela sua aprendizagem, o que faz ele aceitar a reprovação e sentir-se inserido no processo de ensino-aprendizagem. No início do semestre é ministrada uma palestra aos alunos sobre o processo de ensino-aprendizagem.
- Despertar no aluno a percepção da evolução do domínio dos conteúdos comparados as suas experiências anteriores. Isto faz com que o aluno se aperceba que aprendeu com as dificuldades passadas, e o que parecia de difícil resolução ele consegue resolver sem grandes dificuldades.
- Inserir-lo em uma monitoria, onde ele passa a ser um disseminador do conhecimento.
- Confrontá-lo com novos desafios práticos.
- Estimular o aluno a “adotar um colega calouro”. Esta atitude tem como objetivo ajudar na inclusão do aluno repetente na nova turma, para que este não seja estigmatizado como “não inteligente”, como alguém que não sabe nada, resgatando a sua auto-estima.
- Apoio pedagógico - os alunos que apresentaram durante o período letivo problemas de relacionamento, social, familiar, de orientação profissional são encaminhados a coordenação pedagógica, que busca conhecer a realidade do aluno e ajudá-los naquilo que lhes é preciso, melhorando em muitos casos o seu aproveitamento escolar.

4. Conclusões

“O processo de ensino-aprendizagem deve proporcionar ao indivíduo condições de realizar a assimilação de novas informações” (Galhardo, 2006). No processo de ensino-aprendizagem são quatro os componentes que se relacionam: professor, aluno, disciplina a ser lecionada e o ambiente. Quando há um desequilíbrio entre uma delas a probabilidade do fracasso escolar acontecer é muito grande. As instituições precisam desenvolver estratégias para trabalhar com alunos repetentes, para que estes não se sintam excluídos do processo e acabem desistindo de uma profissão sonhada.

O ensino nas disciplinas de Algoritmos e Programação pode ser considerado como alfabetização na Programação de sistemas computacionais. Como tal, deve ser muito bem conduzida, e a primeira experiência dos alunos na grande maioria não é positiva.

O ato de ensinar alguém a descrever seu raciocínio lógico por meio de comandos é desafiador ao professor e ao aluno, e o desafio aumenta quando um professor encontra em uma turma alunos repetentes. A estratégia adotada nas IES em buscar reverter um quadro de reprovação em um sucesso tem se mostrado eficaz. Em torno de 70% dos alunos reprovados se rematriculam novamente na disciplina. Destes, 80% são aprovados na segunda experiência.

Referências

- Ausubel, D. P.; Novak, J. D.; Hanesian, H. (1980). *Psicologia Educacional*. Rio de Janeiro: Editora Interamericana.
- Code::Blocks. (2008) “Code::Blocks”. Disponível em: <http://www.codeblocks.org>. Acesso em: 17 ago. 2008.
- Falkembach, G. A. M.; Araújo, F. V. (2004) “Ensino de Algoritmos utilizando a estratégia ascendente de resolução de problemas com o apoio do ambiente A4”. In: 6º Simpósio Internacional de Informática Educativa, 2004, Cáceres-Espanha.
- Goleman, D. (1999) *Inteligência Emocional*. 36 ed. Rio de Janeiro: Objetiva. ISBN: 8573020806.
- Galhardo, M. A. (2006). Alto índice de reprovação: Como resolver essa situação? Disponível em: http://www.esags.edu.br/biblioteca/colunistas_mariagalhardo20060315.html
- Hostins, H. (2006) “Uma ferramenta para apoio a construção e teste de algoritmos”. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade do Vale do Itajaí, Itajaí.
- Mayer, R. E. (1988) “Introduction to Research on Teaching and Learning Computer Programming”. In: Mayer, R. E. (org.). *Teaching and Learning Computer Programming: Multiple Research Perspectives*, Lawrence Erlbaum Associates.
- Menezes, C. S.; Nobre, I. A. M. (2002) “Um ambiente cooperativo para apoio a cursos de introdução a programação”. In: Congresso da Sociedade Brasileira de Computação, 22, Florianópolis, 2002. Anais... Florianópolis: UFSC, 2002. 1 CD-ROM.

- NetBeans. (2008) “Welcome to NetBeans”. Disponível em: <<http://www.netbeans.org>>. Acesso em: 10 ago. 2008.
- Novakoski, H. J. S/d Reprovação Escolar. Disponível em: <http://br.geocities.com/hermes_filosofia/reprovacao> Acesso em: ago. 2008.
- Pacheco J. A.; Rasche, F.; Canto, C. R. De L. (2007) “Formando competência pela integração interdisciplinar e transdisciplinar”, IX Jornada Catarinense de Educação Tecnológica, Florianópolis, 2007.
- Raabe, A. L. A.; Silva, J. M. C. (2005) “Um Ambiente para Atendimento as Dificuldades de Aprendizagem de Algoritmos”. In: Workshop de Educação em Computação, 13, 2005. Anais. São Leopoldo: UNISINOS, 2005. 1 CD-ROM.
- Rodrigues Jr., M. C. (2004) “Experiências positivas para o ensino de algoritmos”. In: Workshop de Educação em Computação e Informática Bahia-Sergipe, 3, 2004, Salvador. Anais... Salvador: 2004. Disponível em: <<http://www.uefs.br/erbase2004/documentos/weibase/Weibase2004Artigo001.pdf>>. Acesso em: jan. 2005.
- Silva, J. M. C.; Raabe, A. L. A. (2008) “Including Affective Student Model in ITS to Teaching Introductory Programming”. In: Workshop Emotional and Cognitive Issues in ITS, 2008, Montreal.
- Soloway, E.; Ehrlich, K. (1984) “Empirical Studies of Programming Knowledge”. In: IEEE Transactions on Software Engineering, vol. SE-10, n. 5, setembro, 1984.
- Tobar, C. M.; Rosa, J. L. G.; Coello, J. M. A.; Pannain, R. (2001) “Uma arquitetura de ambiente cooperativo para o aprendizado de programação”. In: Simpósio Brasileiro de Informática na Educação, 12, Vitória, 2001. Anais... Vitória: Universidade Federal do Espírito Santo, 2001.