

NewProg - um ambiente *online* para crianças aprenderem programação de computadores

Carpegieri Torezani, Lucinéia Barbosa da Costa Chagas, Orivaldo de Lira Tavares

Programa de Pós-graduação em Informática – Universidade Federal do Espírito Santo (UFES)

carpegieri@gmail.com, cliklucineia@gmail.com, tavares@inf.ufes.br

Abstract. This paper presents the environment *NewProg* created specifically to support the process of learning programming in elementary school children. This work shows the results of a research on the use of this environment for children aged five to eight years. These results show that the environment *NewProg* favors the development of cognitive skills needed for problem solving, promoting authorship and critical thinking in children.

Resumo. Este artigo apresenta o ambiente *NewProg*, criado especificamente para apoiar o processo de aprendizagem de programação em crianças do ensino fundamental. Este trabalho mostra os resultados de uma pesquisa sobre o uso desse ambiente por crianças da faixa etária de cinco a oito anos. Esses resultados mostram que o ambiente *NewProg* favorece o desenvolvimento de habilidades cognitivas necessárias à resolução de problemas, promovendo a autoria e o espírito crítico nas crianças.

1. Introdução

A aprendizagem de programação é benéfica para o desenvolvimento de habilidades cognitivas necessárias para a resolução de problemas. De acordo com Fessakis et alii (2013), a aprendizagem de programação permite o desenvolvimento de conhecimentos em diversas áreas.

O *NewProg* é um ambiente *web* desenvolvido em 2013 por Carpegieri Torezani, com o propósito de auxiliar crianças, da faixa etária de cinco a oito anos, na aprendizagem inicial de programação de computadores. Seu funcionamento se dá em qualquer dispositivo que tenha um navegador *web*, proporcionando uma sensação de entretenimento, enquanto favorece o desenvolvimento de habilidades cognitivas importantes em vários momentos da vida cotidiana, tais como, planejamento e resolução de problemas.

Ao fazer uso desse ambiente, o professor pode acompanhar os avanços/resultados individuais de cada aluno, por meio de relatórios sobre o histórico das soluções elaboradas pelo aluno. O professor é responsável pela escolha de cada atividade e do momento de evolução para outras atividades, de acordo com o desempenho de cada aluno.

O objetivo deste estudo é apresentar o ambiente *NewProg* e resultados iniciais que evidenciam o potencial desse ambiente em favorecer o desenvolvimento, em

crianças da faixa etária de cinco a oito anos, de habilidades cognitivas necessárias para a resolução de problemas, por meio da programação de sequências de ações. Esse ambiente coleta dados sobre as soluções construídas pelas crianças, como evidências das habilidades cognitivas delas na resolução de problemas, para facilitar a análise do professor sobre o desempenho de cada criança e a decisão sobre a proposta da próxima atividade mais adequada para cada caso.

Este artigo está organizado da seguinte forma: na seção 2 apresentamos o referencial teórico; na seção 3 mostramos a metodologia usada; na seção 4 descrevemos a arquitetura do *NewProg*; na seção 5 mostramos os experimentos e resultados obtidos e na seção 6 apresentamos as conclusões seguidas das referências bibliográficas.

2. Referencial teórico

Tavares et alii (2012) relatam que após tentativas frustradas de construir programas para resolver problemas, o aluno reinicia suas ações em busca de uma solução correta, tentando descobrir e corrigir os erros cometidos, inicialmente, em um processo de tentativa e erro, até que consiga ganhar experiência nesse processo que lhe permita tomar consciência sobre o mesmo e sobre a construção de soluções corretas. Esse processo de tentativa e erro é mais curto ou mais longo de acordo com a experiência prévia do estudante.

Relatam ainda que corrigir os erros se torna fundamental para o sucesso do aluno. Nesse processo de corrigir os erros é bem vinda a existência de mecanismos de autorregulação para indicar como aperfeiçoar a solução construída. Ramozzi-Chiarottino (1987) relata que essas autorregulações permitem ao aluno alcançar a capacidade de resolver problemas em nível formal.

Fessakis et alii (2013) apontam que uma das tentativas mais interessantes no uso da programação com crianças é o de Perlman, desenvolvedor do TORTIS, em 1976. TORTIS é um sistema constituído por um conjunto de objetos tangíveis usados para a programação de dispositivos. Seu objetivo é superar a barreira da composição de programas, tornando os benefícios da aprendizagem de programação acessível a crianças a partir de três anos de idade.

Pesquisas enfatizando o uso da programação com crianças têm crescido surpreendentemente, todavia os ambientes existentes como ToonTalk, Squeak Etoys, Scratch, permitem apenas a criação e execução de atividades, não sendo possível um acompanhamento detalhado da evolução dos alunos.

O ambiente *NewProg* proposto neste artigo coleta dados sobre a evolução da aprendizagem dos estudantes, de forma que o professor possa verificar o momento em que acontecem os erros e quais foram os avanços de cada aluno.

O acompanhamento da aprendizagem é feito por relatórios gerados pelo ambiente, onde o professor tem acesso a todas as ações (comandos) selecionadas pelo aluno na tentativa de resolver cada atividade (problema) apresentada. Além disso, o ambiente permite ao professor acompanhar todo o histórico das tentativas do aluno em resolver cada problema.

3. Metodologia

A falta de domínio de técnicas de resolução de problemas faz com que seja difícil de aprender programação. Dessa forma, a fim de auxiliar no desenvolvimento de habilidades cognitivas de ordem superior, adotamos a cooperação como abordagem pedagógica.

Piaget (1936) afirma que "... a cooperação é essencial para o desenvolvimento da razão. Método de trabalho em equipe parece basear-se em mecanismos essenciais da psicologia da criança.". Para Vygotsky (1996) Zona de Desenvolvimento Proximal (ZDP) é a distância entre o nível de desenvolvimento real, determinado pela capacidade de resolver problemas independentemente e o nível de desenvolvimento proximal, demarcado pela capacidade de solucionar problemas com a ajuda de um parceiro mais experiente. Neste sentido, ao iniciarmos as atividades, apresentamos o funcionamento do ambiente NewProg aos quatro professores e aos vinte alunos das séries iniciais com idade entre cinco e oito anos, para familiariza-los com os recursos desse ambiente.

Em seguida, propusemos uma atividade na qual o aluno devia desenvolver individualmente uma sequência de comandos visuais (setas) para resolver um problema. A Figura 01 apresenta um exemplo de sequência de ações.



Figura 01 – Sequência de Ações

No primeiro momento o aluno desenvolvia a atividade de forma individual. Caso não conseguisse um desempenho satisfatório, partíamos para um segundo momento, no qual os estudantes poderiam solicitar o auxílio de algum colega. Todavia o colega podia somente dar algumas dicas para que o estudante conseguisse desenvolver a tarefa.

4. Ambiente NewProg

O *NewProg* é um ambiente de aprendizagem de programação direcionado a crianças de séries iniciais do ensino fundamental. As atividades propostas nesse ambiente priorizam o desenvolvimento de habilidade de resolução de problemas.

O ambiente provê atividades de resolução de problemas de vários níveis de complexidade. Em cada nível de atividade são propostos problemas diferenciados, com o objetivo de ajudar as crianças na resolução de problemas complexos e estimular o raciocínio lógico. A Figura 02 mostra uma tela do *NewProg*.

A Figura 02 exibe a tela inicial do Módulo Cliente, onde o aluno escolhe um avatar que será desafiado a resolver um problema.

O ambiente também possui uma interface inicial onde o usuário tem acesso às seguintes opções:

Padrão - carrega a tela inicial do ambiente.

Andar – permite a edição da sequência de ações que resolve o problema. Permite programar as ações a serem executadas. Uma linguagem visual, com setas de

direção, é usada na composição da sequência de ações necessária para a resolução de cada problema.

Remover Última Ação - remove a última ação da sequência de ações que está sendo editada.

Resultado – apresenta informações ao aluno sobre a execução de cada programa. Esse *feedback* é dado de acordo com o resultado da execução de cada programa. Se o programa estiver correto, o aluno recebe os parabéns pelo seu feito. Se a solução não estiver correta, os erros são indicados e o aluno é incentivado a tentar novamente.

Ajuda – apresenta uma grade (reticulado) na tela durante 10 segundos, de modo a facilitar a localização dos objetos na tela, com o objetivo de auxiliar o aluno a encontrar os possíveis caminhos para concluir a atividade.

Número de Ajudas – informa quantas vezes o aluno fez uso da função ajuda.

Número de Erros – apresenta a quantidade de movimentos inadequados na execução do programa.

Números de Passos – apresenta a quantidade de passos executados pelo avatar no trajeto programado.



A Figura 02 - Tela do ambiente **NewProg**

4.1 Arquitetura NewProg

A arquitetura do ambiente NewProg, representada na Figura 03, é composta de dois módulos: o módulo *Cliente* e o módulo *Servidor*. O módulo *Cliente* oferece uma interface para o usuário ter acesso aos problemas propostos no ambiente e resolvê-los, por meio do uso das funções que permitem a edição do programa. Esse módulo também coleta os dados dessa interação e os envia para o módulo servidor. Esse módulo é carregado pelo navegador *web*, funcionando no dispositivo do usuário, para a execução dos programas desenvolvidos.

O módulo *Servidor* é responsável pelo recebimento e armazenamento dos dados recebidos do módulo Cliente e pela administração do sistema.

Nessa arquitetura o módulo *Servidor* é ligado diretamente ao banco de dados, as funções administrativas, as funções pedagógicas e ao módulo do cliente. O administrador fica responsável pelo cadastro, exclusão e relatórios de turmas e professores. Os professores são responsáveis pelo cadastro e exclusão de alunos (usuários) e tratamento dos relatórios sobre o uso do ambiente pelos alunos. A figura 03 ilustra o funcionamento da arquitetura *NewProg*.

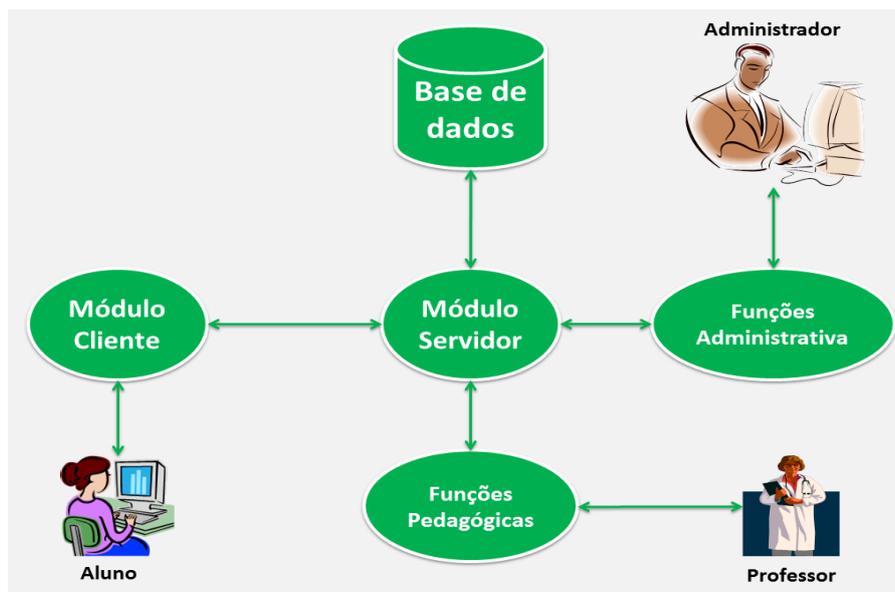


Figura 03 – Arquitetura modular do ambiente *NewProg*

O ambiente *NewProg* usa tecnologia *web* e foi implementado em linguagens *HTML*, *CSS*, *PHP* e *Javascript*.

5. Experimentos e resultados

Os experimentos foram realizados em uma instituição de ensino fundamental com vinte crianças na faixa etária de cinco a oito anos e quatro professores. Na atividade inicial o aluno precisava programar uma sequência de ações de modo a fazer um avatar se locomover na tela do computador. A sequência de ações de cada programa é editada por meio de cliques em setas que correspondem à direção a ser seguida, conforme mostra a Figura 04.

Na segunda atividade, o ambiente apresentava uma tela com obstáculos a serem contornados pelo avatar escolhido. Os alunos precisavam construir uma sequência de ações a serem executadas para fazerem o avatar se locomover até um posto de alimentação. A Figura 05 ilustra a atividade desenvolvida.

Para o desenvolvimento das atividades foram usados os seguintes critérios:

- todos os alunos teriam o mesmo número de tentativas;
- caso o aluno não tivesse sucesso na atividade, em um próximo momento poderia pedir ajuda a um colega de sua escolha;
- obtendo sucesso na atividade, deveria propor um novo caminho.

- o ambiente permite no máximo nove ações a serem incluídas em cada sequência (programa).



Figura 04 – Primeira Atividade



Figura 05 – Segunda Atividade

No início da primeira atividade, notamos que os alunos estavam tendo dificuldades em movimentar o avatar. Neste momento, eles fizeram alguns questionamentos como: “Esta atividade é um jogo?”, “Porque o bichinho não mexe a cada indicação de ação?”.

Após uma breve explicação do funcionamento da atividade, comunicamos aos alunos que o avatar se movimentaria quando as ações (setas) escolhidas fossem

executadas. Após essa explicação, os alunos começaram a escolher os comandos e clicar em executar. Em seguida o ambiente executava as ações programadas pelos alunos.

O experimento foi feito durante o segundo bimestre de 2013, com vinte crianças de uma escola da rede municipal, da faixa etária de cinco a oito anos, sendo cinco crianças de cada idade. A Tabela 1 mostra os resultados obtidos durante a pesquisa. A cada execução coletávamos os dados de cada criança e questionávamos se estavam animados a continuar.

Tabela 1 – Média dos dados Primeira Atividade.

	Primeira amostragem	Segunda amostragem	Terceira amostragem	Quarta amostragem
Êxito	12%	35%	70%	95%
Erros	5	4	2	1
Motivação	Baixa	Média	Boa	Alta

Na implementação da segunda atividade, foi explicado aos alunos que o grau de complexidade seria maior do que na atividade anterior, porém os princípios seriam os mesmos. A Tabela 2 apresenta os resultados obtidos durante essa etapa da pesquisa.

Destacamos que houve um aumento significativo no número de erros, pois, devido à possibilidade de algumas ações executarem vários passos, alguns alunos apresentaram dificuldades para a correta programação do caminho.

Tabela 2 – Média dos dados Segunda Atividade.

	Primeira amostragem	Segunda amostragem	Terceira amostragem	Quarta amostragem
Êxito	30%	55%	85%	98%
Erros	9	6	3	2
Motivação	Média	Boa	Alta	Alta

Durante o período em que foi realizada a pesquisa, buscamos procurar evidências que comprovassem a aceitação do ambiente *NewProg* pelos alunos e professores. A Figura 06 exibe a avaliação do ambiente por parte de alunos e professores. A cada etapa do processo questionávamos os participantes sobre o uso do ambiente, indagando-os se concordavam com seu uso e se percebiam benefícios.

A Figura 06 evidencia ainda os avanços alcançados no desempenho dos alunos através dos experimentos. Notamos que o desempenho no uso do ambiente por parte dos usuários (alunos e professores) aumentava gradativamente. Ao final da pesquisa observamos que os alunos elaboravam sequências de ações corretas e eficientes (quantidade de ações próxima da solução de caminho mínimo).

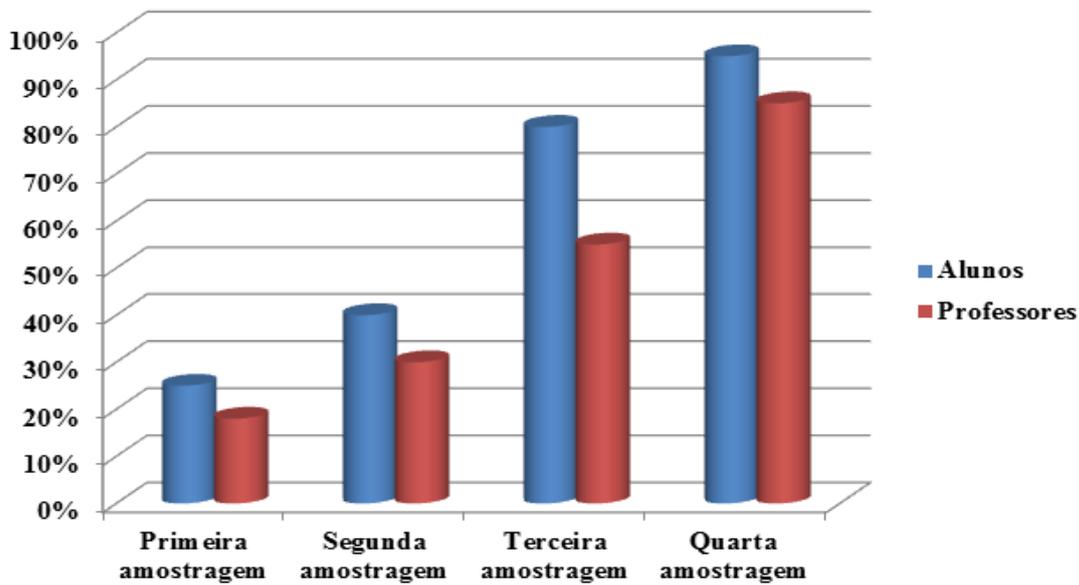


Figura 06 – Avaliação do ambiente *NewProg*

5.1 Avaliação da aprendizagem

Durante a pesquisa, buscamos os indicadores que comprovassem que realmente houve a aprendizagem de programação e ganhos cognitivos por parte das crianças. Chegamos aos seguintes indicadores: identificação e correção de erros, busca por várias soluções para a resolução dos problemas, reflexão sobre as soluções propostas, autoria e espírito crítico. Esses indicadores foram levantados através de observações durante o experimento e análise dos relatórios gerados pelo ambiente.

O uso desses indicadores na análise dos dados coletados durante o desenvolvimento das atividades pelas crianças, mostra que houve, ao longo do uso do ambiente, significativa evolução dos mesmos, comprovando a aprendizagem de programação e a evolução de habilidades cognitivas.



Figura 07 – Execução da Segunda Atividade

A sequência de Figuras 07, 08 e 09 mostra as tentativas de solução da Segunda Atividade por uma das crianças de oito anos de idade.

Na Figura 07 o objetivo de levar o avatar ao alimento foi alcançado, mas a sequência de ações contém 5 erros que são sinalizados na interface do *NewProg*. A criança recebe os parabéns do sistema e dos professores e é incentivada a buscar uma nova solução sem erros.



Figura 08 – Execução da Segunda Atividade

A Figura 08 mostra uma nova solução construída pela mesma criança, em que o objetivo foi cumprido sem erros. Após isso, os professores perguntam a criança se ela consegue elaborar uma solução com um menor número de ações.



Figura 09 – Execução da Segunda Atividade

A figura 09 mostra uma nova solução construída pela criança, em que o objetivo foi alcançado, sem erro, e com uma quantidade menor de ações, ou seja, uma solução mais eficiente do que as duas anteriores.

O exemplo de evolução da criança na construção de soluções mais eficientes para um problema, apresentado na seção 5.1, evidencia o potencial do ambiente *NewProg* em ajudar crianças a aprenderem conceitos de programação, enquanto visualizam a execução das ações escolhidas por elas. Ao mesmo tempo em que aprendem a programar, desenvolvem habilidades cognitivas como abstração, análise crítica das soluções construídas por elas ou pelos colegas, comparação de soluções e a colaboração que é exercitada quando veem a solução do colega e tiram dúvidas com os educadores ou com os colegas. Por outro lado, os educadores ganham uma ferramenta muito importante para motivar as crianças a desenvolverem essas habilidades cognitivas.

6. Conclusões

Apresentamos o ambiente pedagógico *NewProg*, construído para auxiliar e acompanhar indivíduos da faixa etária de cinco a oito anos, durante o processo de aprendizagem de programação de computadores.

Os resultados apresentados mostram que o ambiente *NewProg* favoreceu o desenvolvimento de habilidades cognitivas necessárias à resolução de problemas, promovendo a autoria e o espírito crítico dos indivíduos observados. O uso do *NewProg* favoreceu avanços no desempenho cognitivo de indivíduos da faixa etária de cinco a oito anos, permitindo ao educador um acompanhamento da evolução das habilidades adquiridas por esses indivíduos no processo de programar.

Este projeto continuará com a ampliação das atividades catalogadas no *NewProg* e o aumento dos dados coletados sobre o desempenho dos indivíduos observados, de modo a facilitar a identificação do desenvolvimento cognitivo deles, nas várias etapas do uso do ambiente.

7. Referências

- Fessakis, G.; Gouli, E.; Mavroudi, E., Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education* 63 (2013) 87–97.
- Piaget, J.; O trabalho por equipes na escola; traduzido para o português por Luiz G. Fieury (Revista de Educação - Diretoria do Ensino do Estado de São Paulo; set/dez, 1936); adaptado para o português moderno por Andrea A. Botelho (Laboratório de Psicopedagogia do Instituto de Psicologia da USP, 1993); *Revista Psicopedagogia*, 15 (36)- 1996.
- Ramozzi-Chiarottino, Z.; *Psicologia e epistemologia genética de Jean Piaget*; Editora Pedagógica e Univesitária Ltda.; ISBN 85-12-62350-0; São Paulo; 1987.
- Tavares, O. L.; Menezes, C.S.; Nevado, R.A.: Pedagogical architectures to support the process of teaching and learning of computer programming: In FIE2012- Frontiers in education conference, 2012.
- Vygotsky, L. S.; *A formação social da mente*. Rio de Janeiro: Martins Fontes, 1996.