

“A Última Árvore”, exercitando o Pensamento Computacional por meio de um jogo educacional baseado em Gramática de Grafos*

Braz Araujo da Silva Junior¹, Simone André da Costa Cavalheiro², Luciana Foss²

¹Laboratório de Fundamentos de Computação - CDTec - Universidade Federal de Pelotas
CEP 96.010-610 - Pelotas - RS - Brazil

²Programa de Pós-Graduação em Computação - Universidade Federal de Pelotas
CEP 96.010-610 - Pelotas - RS - Brazil

{badsjunior, simone.costa, lfoss}@inf.ufpel.edu.br

Abstract. *This paper presents a graph grammar based educational game as an alternative approach to the computational thinking development. It is about a turn-based strategy boardgame wherein the students play animals restoring a nearly destroyed forest. It is highlighted the stimulation of the skills: data collection, analysis and representation; parallelization; and abstraction.*

Resumo. *Este artigo apresenta um jogo educacional baseado em gramática de grafos como uma abordagem alternativa para o desenvolvimento do pensamento computacional. Se trata de um jogo de tabuleiro de estratégia baseada em turnos onde os alunos interpretam animais para restaurar uma floresta quase totalmente destruída. É destacado o estímulo das habilidades: coleta, análise e representação de dados; paralelismo; e abstração.*

1. Introdução

Desde que foi apresentado à comunidade acadêmica em [Wing 2006], o Pensamento Computacional (PC) tem recebido atenção de educadores ao redor do mundo. Wing argumenta que os avanços na computação possibilitaram pesquisadores dentre todas as disciplinas visionar novas estratégias de resolução de problemas e testar novas soluções tanto no mundo virtual quanto no real [Barr and Stephenson 2011]. Para consolidar o termo, a *Computer Science Teachers Association* (CSTA), a *International Society for Technology in Education* (ISTE) e a *National Science Foundation* (NSF) estabeleceram sua definição operacional. O PC é um processo de resolução de problemas que inclui (mas não é limitado a) as seguintes características: formular problemas de uma maneira que nos permita usar um computador ou outras ferramentas para ajudar a resolvê-los; organizar e analisar dados logicamente; representar dados através de abstrações, como modelos e simulações; automatizar soluções através do pensamento algorítmico (uma série de passos ordenados); identificar, analisar e implementar possíveis soluções com o objetivo de alcançar a mais eficiente e efetiva combinação de passos e recursos; e generalizar e transferir este processo de solução de problemas para uma ampla variedade de problemas [CSTA et al. 2011].

*Projeto com chancela da SBC e realizado com o apoio da PREC e PRPPG / UFPel e CNPq (UNIVERSAL 447463/2014-4).

Uma revisão sistemática da literatura identificando as principais ferramentas que foram utilizadas no ensino-aprendizagem do PC de 2006 até o início de 2015 é apresentada em [Bombasar et al. 2015]. Neste levantamento é evidenciado que ferramentas como *Scratch*, *Alice* e *CS Unplugged* são abordagens comuns para o desenvolvimento do PC. Diversos esforços têm sido realizados para exercitar as habilidades do PC. São exemplos de diferentes abordagens que têm sido adotadas para o desenvolvimento delas: o uso de jogos digitais [Pinho et al. 2016, Falcão and Barbosa 2015]; a robótica [de Souza et al. 2016, Oliveira and Araujo 2016]; a proposição de atividades e jogos não-digitais [Andrade et al. 2013]; e o ensino de programação (muitas vezes adotando linguagens visuais) [Rodriguez et al. 2015].

Gramática de Grafos (GG) é uma linguagem visual que permite a representação de um problema e/ou sua solução usando um alto grau de abstração de dados e processos, onde os estados de um sistema são descritos por grafos e as mudanças de estados são definidas por regras de transformação de grafos. Ao observar relações entre habilidades fundamentais do PC e características das GGs, como análise de dados, simulação e paralelismo, este artigo propõe uma abordagem alternativa para a promoção do PC por meio de um jogo educacional fundamentado em GG. O jogo proposto denomina-se **A Última Árvore** e consiste em um jogo de tabuleiro de estratégia baseada em turnos onde os jogadores devem tentar restaurar uma floresta destruída, na qual restou somente uma árvore. Nesta jornada cada jogador deve analisar os elementos sobre o tabuleiro a fim de satisfazer as condições das ações que precisam tomar, desenvolvendo as habilidades de coleta, análise e representação de dados. As ações e suas condições são descritas por grafos, elevando o nível de abstração. Embora todos os jogadores colaborem para a restauração da floresta, cada um sorteia um objetivo particular no início da partida, desta forma os jogadores lidam com o paralelismo e concorrência.

Na concepção e desenvolvimento do jogo proposto, adotou-se a estratégia desplugada, na qual não são necessários computadores ou recursos tecnológicos adicionais para a sua implementação [Bell et al. 2009]. Esta estratégia tem sido usualmente adotada [Bombasar et al. 2015] e traz como principal vantagem a facilidade de implementação, já que não é necessário dispor de laboratórios de informática para ser aplicada.

O artigo está organizado como segue. A seção 2 expõe a metodologia de concepção e desenvolvimento do jogo. A seção 3 faz uma breve descrição do formalismo GG e apresenta a gramática que define o jogo. A seção 4 apresenta uma conclusão e a discussão sobre trabalhos futuros.

2. Metodologia de Concepção e Desenvolvimento

Para a concepção e o desenvolvimento do jogo educacional *A Última Árvore* utilizou-se a metodologia ENGAGED (do inglês *Educational Games Development*), proposta por [Battistella and von Wangenheim 2016]. A Tabela 1 traz uma versão resumida da visão geral do processo de elaboração ENGAGED aplicado ao jogo, explicitando as 3 fases envolvidas no seu desenvolvimento.

Na Fase 1, análise da Unidade Institucional (UI), os objetivos, contexto, pré-requisitos e público alvo do jogo foram especificados. A caracterização dos aprendizes (A1.2) foi feita considerando o desenvolvimento gradual do jogo. Optou-se por iniciar com um jogo mais simples e fácil de se jogar e, futuramente, construir sobre esta base

novas funcionalidades de maior complexidade. Deste modo, o público alvo inicial foi definido como os alunos do ensino fundamental. Os objetivos de desempenho (A1.3) foram formulados a partir do conjunto de habilidades do PC bem como das competências necessárias para a manipulação de uma GG. Na Fase 2, projeto da UI, foram estipulados os conteúdos envolvidos e as estratégias de avaliação.

Tabela 1. Visão geral do processo de elaboração através do modelo ENgAGED

Fase 1. Análise da Unidade Instrucional (UI)	
A1.1 Especificar UI do jogo	Objetivo: desenvolvimento de competências do PC, em particular, análise e manipulação de dados. Pré requisito: noções básicas de lógica.
A1.2 Caracterizar aprendizes	Público-alvo: alunos do ensino fundamental.
A1.3 Definir objetivo(s) de desempenho	Os alunos devem ser capazes de: realizar a coleta, análise e manipulação de dados; decompor e abstrair problemas; traçar algoritmos ; e simular eventos. Também devem estar aptos a: identificar elementos (vértices) e suas relações (arestas) num grafo; analisar as condições de aplicações de regras de uma GG; aplicar regras corretamente; perceber concorrência ou paralelismo entre regras; e projetar as modificações que cada aplicação desencadeia no sistema.
Fase 2. Projeto da Unidade Instrucional (UI)	
A2.1 Definir avaliação do aluno	As avaliações devem ser realizadas ao final de cada partida. Observa-se a distância entre a condição do objetivo recebido por cada aluno e o estado final do jogo. Discutem-se as estratégias tomadas e justificativas para tomá-las, dadas pelos alunos.
A2.2 Definir conteúdo da estratégia instrucional	A estratégia instrucional é um jogo educacional baseado em GG para desenvolver o PC . Nele, deve ser manipulada uma GG, realizando aplicações de regras para alcançar um objetivo.
A2.3 Decidir pelo desenvolvimento ou utilização	Optou-se pelo desenvolvimento já que não se conhece uma proposta similar de jogo que possa ser utilizada.
A2.4 Revisar o modelo de avaliação do jogo	Para avaliar o desempenho dos alunos e eficiência do projeto, adotou-se o modelo de avaliação em 4 níveis de [Kirkpatrick 1994]. Nele são propostos métodos de avaliação sobre: a reação ao processo de treinamento; o aprendizado , diferença do conhecimento antes e depois; o comportamento , a aplicação dos conceitos aprendidos; e o resultado , o impacto de ter adquirido o conhecimento.
Fase 3. Desenvolvimento do Jogo Educacional	
Fase 3.1 Análise do Jogo - Requisitos	Requisitos de aprendizagem: descritos na Tabela 2. Requisitos para o desenvolvimento do PC: detalhados na Tabela 3.
Fase 3.2 Concepção do Jogo	A Última Árvore é uma representação lúdica de uma GG. Nele os jogadores interpretam animais de uma floresta recém desmatada onde há uma única árvore restante. Os objetivos de cada jogador são sorteados no início de cada partida. Somente o próprio jogador tem conhecimento de seu objetivo. Mas todos eles norteiam a restauração da floresta, por exemplo: “Um lugar deve possuir 5 plantas ao mesmo tempo”. Para alcançar os objetivos, os jogadores simulam o processo natural de expansão de uma floresta , como o ciclo de crescimento das árvores e a disseminação das sementes por agentes externos (os animais). Em seu turno, cada jogador realiza uma ação: mover, colher, plantar, regar, adubar, ou comer . Vence quem alcançar primeiro o objetivo sorteado.
Fase 3.3. Design do Jogo	
A3.3.1 Definir engine	A Última Árvore deve corresponder a especificação de uma GG .

A3.3.2 Produzir ilustrações ou imagens dos elementos do jogo	Nesta fase foram geradas as artes dos produtos . A unidade de A Última Árvore é composta por: um manual (não precisa ser impresso); três folhas de tabuleiro ; duas folhas de regras ; uma folha de itens; e um conjunto de cartas , sendo quatro animais, quatro setas, dez sementes, dez plantas, dez árvores e dez frutas. Todo o material foi produzido de forma a facilitar sua implementação, sendo projetados para serem impressos em folhas A4 , exigindo somente alguns recortes nos lugares indicados.
A3.3.3 Modelar o jogo	Modelagem feita de forma automatizada. Por ser uma GG, foi utilizada a ferramenta G Raphs for O bject- O riented V erification (GROOVE) [GROOVE 2017], viabilizando a especificação e simulação da GG definida. Foi possível simular a execução do jogo, explorando automaticamente todas as possibilidades de aplicação de regras, e registrando todos os passos de derivação utilizados para se alcançar os diferentes estados finais.
Fase 3.4. Implementação do Jogo	Impressão e montagem dos elementos. A Figura 1 exibe protótipo do jogo, impresso para testes.
Fase 3.5 Testes do jogo	Testes foram realizados com os criadores do jogo e “especialistas”, usuários com conhecimento prévio acerca de linguagens formais: alunos de graduação em engenharia da computação e ciência da computação.

Na primeira etapa da Fase 3, desenvolvimento do jogo, definiu-se os requisitos de aprendizagem e de desenvolvimento do PC. Os requisitos de aprendizagem foram especificados considerando o espectro cognitivo relacionado à dimensão do conhecimento proposto pela Taxonomia de Bloom Revisada. A Taxonomia de Bloom clássica divide o processo cognitivo em uma hierarquia de 6 níveis (da base para o topo): **conhecimento, compreensão, aplicação, análise, síntese e avaliação**. Sua versão revisada, proposta em [Anderson et al. 2001], altera esta divisão, removendo “síntese” e incluindo novos níveis: **lembrança**, na base da hierarquia; e **criação**, no topo. Além disso, o nível do **conhecimento** é relacionado à todos os outros e dividido em 4 tipos: **factual**, de detalhes e elementos específicos; **conceitual**, de classificações, generalizações e modelos; **procedural**, de técnicas, métodos e algoritmos; e **metacognitivo**, sobre o próprio processo cognitivo e conhecimento. Para atingir um aprendizado consistente ao jogar, foram buscadas ações que explorassem cada nuance do processo cognitivo relacionada a cada forma de conhecimento, como detalhado na Tabela 2. Já os requisitos para o desenvolvimento do PC, expostos na Tabela 3, foram delineados a partir da inter-relação entre conceitos do PC e a especificação de uma GG.

Na segunda etapa da Fase 3, foi feita a concepção, modelagem e implementação de A Última Árvore. Optou-se por uma atividade desplugada para evitar as dificuldades que laboratórios de informática enfrentam, como despreparo dos profissionais e falta de infraestrutura [da Silva and Santos 2017]. O jogo corresponde a especificação de uma GG, apresentada na seção 3 deste artigo.

Um protótipo do jogo¹ é ilustrado na Figura 1. O tabuleiro do jogo é composto por uma matriz 2x3 de folhas A4. A primeira linha tem as ações (regras) ilustradas nos cantos e o painel dos personagens no centro. A segunda linha possui o mapa, que contém: os blocos de terra; os caminhos entre eles; e três espaços marcados para cada bloco de terra. Eles indicam onde devem ser colocadas as cartas de cada tipo quando aquele bloco de terra receber um elemento daquele tipo. Por exemplo, o destaque 3 indica onde devem ser colocadas as cartas de planta quando uma planta for criada naquele bloco de terra.

¹Para o download do protótipo do jogo acesse <https://goo.gl/jX9dTV>



	Factual	Conceitual	Procedural	Metacognitivo
Lembrar	Listar mentalmente as noções básicas e intuitivas de lógica	Reconhecer os padrões presentes na definição de cada ação (regra)	Relembrar como funcionam estruturas lógicas, como condição e consequência	Identificar , nos padrões reconhecidos, conexões com a lógica lembrada
Entender	Resumir conjuntos de ações em processos que tenham como saída o elemento alvo	Classificar as ações que podem ser tomadas	Elucidar-se quanto ao impacto das ações sobre o objetivo	Prever a interpretação dos outros sobre as suas jogadas e objetivos
Aplicar	Responder às alterações dos elementos correntes no jogo	Fornecer situações ou propícias aos outros jogadores	Executar uma ação que ajude a atingir o seu objetivo ou atrapalhe o dos outros	Usar as ações dos outros jogadores para alcançar os seus objetivos
Analisar	Selecionar a melhor ação dada a situação corrente no jogo	Diferenciar as ações que visam alcançar objetivo das que visam impedir um	Integrar os elementos correntes no jogo aos objetivos e ações possíveis	Desconstruir hipóteses dos outros simulando tentar alcançar outro objetivo
Avaliar	Monitorar as ações dos outros jogadores e seus progressos	Determinar as consequências de cada ação no progresso geral	Julgar a estratégia e objetivo dos outros por suas ações	Refletir sobre o progresso de todos e as distâncias de seus objetivos
Criar	Gerar , no jogo, elementos que façam parte do seu objetivo	Montar uma estratégia através de um conjunto de ações	Planejar ações alternativas para se adaptar a diferentes situações	Produzir , no jogo, todos os elementos de seu objetivo

Tabela 3. Design do jogo considerando as relações do PC com GG

Habilidade do PC	Relação com a Gramática de Grafos	Implicações no Design
Coleta de Dados	Ser uma linguagem visual permite a extração de dados de maneira direta.	Todos os elementos do jogo devem ter uma representação gráfica .
Análise de Dados	Uma linguagem formal permite verificações de propriedades.	O jogo inteiro deve ser formalmente descrito como uma GG .
Representação de Dados	GGs resumem sistemas em: grafo-tipo, conjunto de regras, e grafo inicial . Eles definem no sistema, respectivamente, os tipos de vértices e arestas permitidos, como ocorrem as transições de estado e o estado inicial.	A disposição dos elementos do jogo deve ser restrita por um grafo-tipo . As ações dos jogadores devem ser descritas por um conjunto de regras . E o tabuleiro deve representar um grafo inicial no início de cada partida.
Decomposição de Problemas	A aplicação de uma regra requer encontrar, no grafo, elementos que satisfazam a condição dela (um <i>match</i>), para então atingir uma consequência	A execução do jogo deve exercitar no jogador a análise de ambos os lados de uma regra, L e R , condição e consequência. Além da habilidade de identificar <i>matches</i> .
Abstração	Abstrai-se a sintaxe tornando ela intuitiva através da substituição dos elementos textuais por grafos	Representação de regras (ações) por grafos, no lugar de sentenças escritas
Algoritmos e Processos	Aplicações de regras representam passos atômicos para a construção de algoritmos	Objetivos devem ser processos que possam ser alcançados através de sequências de ações (regras da GG)
Simulação	A aplicação de regras a partir do estado inicial simula os comportamentos possíveis de um sistema	O jogo é especificado como um sistema com comportamentos bem definidos e facilmente reconhecíveis
Paralelismo	Várias regras podem ser executadas ao mesmo tempo , desde que não deletem/modifiquem os mesmos elementos no grafo alvo	A concepção das regras deve ser feita de maneira que duas ou mais ações não concorram a nenhum recurso entre si , permitindo o paralelismo.

Os tipos dos elementos de cada carta do jogo são indicados pelas cores de suas bordas, divididas em: sementes (marrom); plantas (verde claro); árvores (verde escuro); frutas (vermelho); e animais (azul). Para cada carta de animal há uma carta de posicionamento correspondente (destaque 1 na Figura 1). As cartas dos animais devem ser colocadas nos espaços correspondentes no painel de personagens (destaque 4), e servirão para indicar a quantidade de frutas que cada jogador possui. Isto é, quando um animal recebe uma fruta, deve-se colocar uma carta de fruta no respectivo espaço indicado. Se já houver(em) carta(s) no local, coloca-se a nova carta no topo da(s) antiga(s). As cartas de posicionamento devem ser colocadas sobre os blocos de terra no mapa (destaque 5), e indicam onde cada animal está. As cartas de borda rosa são os objetivos individuais (destaque 2), as quais devem ser embaralhadas no começo de cada partida, e então sorteadas uma para cada jogador.

Durante os testes do jogo na Fase 3.5 (última etapa realizada), foi requisitado aos participantes que respondessem a um questionário onde podiam descrever sua visão geral, prós, contras e sugestões. Destas respostas, foram ressaltadas características do protótipo: **intimidação**, num primeiro momento o jogo não parece tão simples quanto é;

generalização, após a compreensão de uma regra as demais são facilmente compreendidas; e **evolução**, a cada nova partida os jogadores refinam suas estratégias e percepção.

3. Especificação da Gramática de Grafos

Uma GG é uma generalização das gramáticas de Chomsky, substituindo *strings* por grafos [Ehrig et al. 1997]. **Um grafo é definido por:** um conjunto de **vértices**, um conjunto de **arestas**, uma função total que define as **origens** das arestas e uma função total que define os **destinos** delas. Um **morfismo entre grafos** G_a e G_b funciona como um mapeamento dos elementos de G_a em G_b , tal que o mapeamento de cada aresta deve preservar (o mapeamento de sua) origem e destino. Um morfismo é **injetor** quando não há dois (ou mais) elementos do G_a sendo mapeados para o mesmo elemento do G_b . E é **total** quando todos os elementos do G_a possuem mapeamento em G_b , caso contrário o morfismo é dito parcial.

Uma GG é composta por um grafo tipo, um grafo inicial e um conjunto de regras. O **grafo tipo** especifica e restringe os elementos (vértices e arestas) possíveis em uma GG. Todos os grafos da GG são tipados sobre o mesmo grafo tipo. Isto é, se G_T é o grafo tipo de uma GG, qualquer grafo G_X que pertença à gramática possui um **morfismo total** de G_X em G_T . O que define o comportamento do sistema em uma GG é o **conjunto de regras**. Uma **regra** $r : G_L \rightarrow G_R$ é composta por dois grafos, o lado esquerdo (G_L) e o direito (G_R), relacionados por um morfismo parcial injetor de G_L para G_R . O lado esquerdo da regra define os elementos que devem estar presentes no grafo estado para que ela possa ser aplicada, e o lado direito descreve o efeito desta aplicação. Os elementos em G_L , mapeados pelo morfismo, devem ser preservados e os não mapeados devem ser deletados, enquanto os elementos em G_R que não são imagem de G_L devem ser criados. Por fim, o **grafo inicial** G_0 de uma GG especifica o estado inicial do sistema.

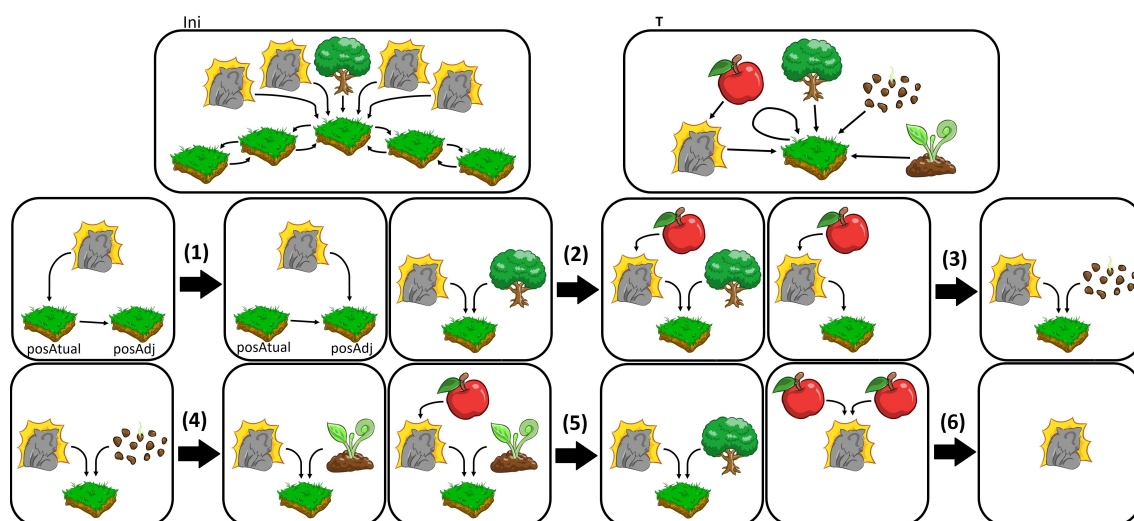


Figura 2. Gramática de Grafos de A Última Árvore

A GG que descreve A Última Árvore é exibida na Figura 2. O grafo inicial Ini descreve a configuração inicial do jogo para 4 jogadores. O grafo tipo T estabelece os tipos de vértices e relações que ocorrem no jogo, definindo que: todas as sementes, plantas e árvores são associadas a um bloco de terra; os blocos de terra podem estar associados

a outros blocos de terra, indicando um caminho; os animais devem estar em algum bloco de terra (não podem estar em uma árvore, por exemplo); e as frutas só podem estar com animais, não em blocos de terra ou em árvores. As regras descrevem as ações que podem ocorrer no jogo: (1) *mover*, especifica a mudança de posição do jogador para uma posição adjacente; (2) *colher*, define a colheita de uma fruta a partir de uma árvore presente na posição corrente do jogador; (3) *plantar*, estabelece a utilização de uma fruta do jogador para enterrar uma semente em sua posição atual; (4) *regar*, descreve a transformação de uma semente em uma planta na posição do jogador; (5) *adubar*, explicita a transformação de uma planta, presente na posição do jogador, em uma árvore utilizando uma fruta (como adubo); (6) *comer*, apresenta o consumo de duas frutas por um mesmo jogador.

Na representação gráfica, usualmente, os morfismos não são representados explicitamente. Assume-se que os itens do grafo esquerdo são mapeados para aqueles que tem o mesmo rótulo ou representação do lado direito.

A semântica de um sistema especificado por uma GG é dada por um sistema de transição que representa as possíveis aplicações de regras a partir do **grafo inicial**. Para aplicar uma regra $r : G_L \rightarrow G_R$ em um grafo G_X , deve existir uma ocorrência (ou *match*) de todos os elementos do grafo G_L no grafo G_X . Esta ocorrência é definida por um morfismo total e injetor. A aplicação de uma regra r sobre um grafo G_X é definida por um *pushout* [Craig et al. 1993] na categoria de grafos e morfismos parciais. Esta operação constrói um novo grafo eliminando de G_X os elementos deletados pela regra e adicionando a G_X os elementos criados pela regra. Como exemplo, a Figura 3 ilustra uma possível aplicação (seguindo a abordagem *single pushout* [Ehrig et al. 1997]) da regra *mover* no estado inicial da GG do jogo.

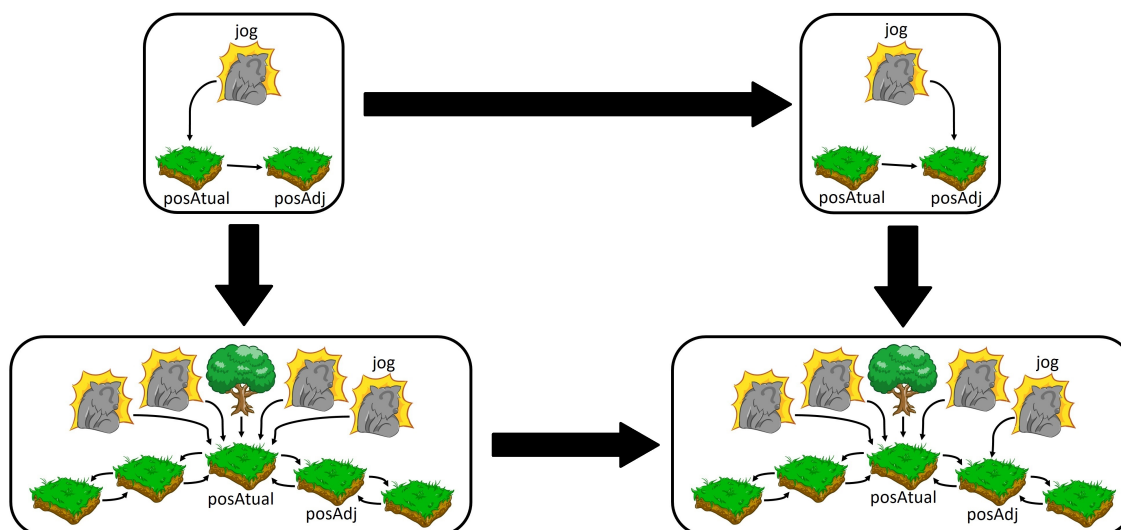


Figura 3. Aplicação da regra *mover* no grafo inicial *Ini* da GG do jogo.

4. Conclusão e Trabalhos Futuros

Neste artigo foi apresentada uma abordagem alternativa para o desenvolvimento do PC, o jogo educacional baseado em GG: A Última Árvore. Como a abordagem mais comum utilizada para o exercício do PC têm sido através de linguagens de programação

[Bombasar et al. 2015], este trabalho se diferencia ao propor exercitar habilidades do PC através da execução de um jogo. Em [Bombasar et al. 2015] é evidenciado a ausência de ferramentas baseadas em teorias e formalismos (então é sugerido a utilização de ferramentas inspiradas na Máquina de Turing). A Última Árvore preenche esta lacuna ao ser uma ferramenta baseada no formalismo GG. Assim, a proposta se firma em explorar as relações entre as GG e as habilidades do PC. Mesmo que o jogador não conheça a definição formal de uma GG, ao jogá-lo acaba simulando a execução de uma GG e respeitando sua especificação. Com isto, trabalha conceitos de computação que exercitam o PC, como: coleta, análise e representação de dados, tendo a todo instante que reavaliar a situação atual do jogo, seus elementos e as relações entre eles; concorrência e paralelismo, ao ter que observar que suas jogadas podem estar ajudando/atrapalhando a alcançar não só seu próprio objetivo, mas também o dos outros; decomposição e simulação, ao reproduzir todos os passos presentes no processo natural de expansão de uma floresta através de um sistema descrito por uma GG; e abstração, ao ser poupado da preocupação com comandos e sintaxes textuais, focando-se em suas funções.

O jogo foi elaborado através do processo ENgAGED, que apresenta duas fases não contempladas neste artigo: 4 e 5, execução e avaliação da UI, respectivamente. Como trabalhos futuros tem-se a conclusão destas fases e extensões do jogo. As extensões incluem: a adição de outros elementos de GG, como a condição negativa de aplicação (NAC, do inglês *Negative Application Condition*) e atributos (redefinindo a GG como uma *Attributed Graph Grammar*); e criação de novos modos de jogo, como modo paralelo, onde todos os jogadores realizam uma ação no mesmo turno e novos níveis de dificuldade, com regras e objetivos mais complexos para promover um aprendizado gradativo.

Referências

- Anderson, L. W. et al. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Longman, New York.
- Andrade, D. et al. (2013). Proposta de atividades para o desenvolvimento do pensamento computacional no ensino fundamental. In *Workshop de Informática na Escola*, pages 169–178.
- Barr, V. and Stephenson, C. (2011). Bringing computational thinking to k-12: what is involved and what is the role of the computer science education community? *Acm Inroads*, pages 48–54.
- Battistella, P. E. and von Wangenheim, C. G. (2016). Engaged: Um processo de desenvolvimento de jogos para ensinar computação. In *Simpósio Brasileiro de Informática na Educação-SBIE*, pages 380–389.
- Bell, T. et al. (2009). Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology*, 13:20–29.
- Bombasar, J., Raabe, A., de Miranda, E. M., and Santiago, R. (2015). Ferramentas para o ensino-aprendizagem do pensamento computacional: onde está alan turing? In *Simpósio Brasileiro de Informática na Educação-SBIE*, pages 81–90.

- Craigen, D., Gerhart, S., and Ralston, T. (1993). An international survey of industrial applications of formal methods. In *Z User Workshop*, pages 1–5.
- CSTA, ISTE, and NSF (2011). *Computational Thinking Leadership Toolkit*. CSTA.
- da Silva, M. N. S. and Santos, M. M. (2017). As tecnologias de informação e comunicação no ambiente escolar. *Revista Educação & Tecnologia*, 15.
- de Souza, I. M. L. et al. (2016). Explorando robótica com pensamento computacional no ensino médio: Um estudo sobre seus efeitos na educação. In *Simpósio Brasileiro de Informática na Educação-SBIE*, pages 490–499.
- Ehrig, H. et al. (1997). *Handbook of graph grammars and computing by graph transformation: volume I. Foundations*. World Scientific Publishing.
- Falcão, T. P. and Barbosa, R. (2015). ”aperta o play!” análise da interação exploratória em um jogo baseado em pensamento computacional. In *Simpósio Brasileiro de Informática na Educação-SBIE*, pages 419–428.
- GROOVE (2017). Graphs for object-oriented verification. Disponível em: <http://groove.cs.utwente.nl/>. Acesso em 30 de Junho de 2017.
- Kirkpatrick, D. (1994). *Evaluating Training Programs: The Four Levels*. Berrett-Koehler.
- Oliveira, E. and Araujo, A. L. (2016). Pensamento computacional e robótica: Um estudo sobre habilidades desenvolvidas em oficinas de robótica educacional. In *Simpósio Brasileiro de Informática na Educação-SBIE*, pages 530–539.
- Pinho, G. et al. (2016). Proposta de jogo digital para dispositivos móveis: Desenvolvendo habilidades do pensamento computacional. In *Simpósio Brasileiro de Informática na Educação-SBIE*, pages 100–109.
- Rodriguez, C., Zem-Lopes, A. M., Marques, L., and Isotani, S. (2015). Pensamento computacional: transformando ideias em jogos digitais usando o scratch. In *Workshop de Informática na Escola*, pages 62–71.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.