

# Star Owners: um modelo construtivista de desenvolvimento de *Serious Games* para jogo de ensino de Teoria da Computação

Matheus dos Santos Luccas<sup>1</sup> e Kalinka Castelo Branco<sup>1</sup>

<sup>1</sup>Instituto de Ciências Matemáticas e de Computação (ICMC)  
Av. Trabalhador São Carlense, 400 – 13.565.590 – São Carlos – SP – Brasil

matheus.luccas@usp.br, kalinka@icmc.usp.br

**Abstract.** *This paper presents COMBO (Ciclo Organizado Modular Baseado em Oportunidades) methodology created to allow the development of constructivist educational games based on the principles of expressiveness and exploitability. COMBO methodology was evaluated and tested through the development of the game Star Owners, which addresses specific concepts that don't require prior knowledge from the student/player. The methodology elaboration proposes to mitigate gaps and common errors observed in initiatives of application and development of educational games and proved to be successful since the development of a game using COMBO was carried out and the same was applied in a course.*

**Keywords.** *Serious Games; Constructivism; Games in Education; Methodology.*

**Resumo.** *Este artigo apresenta a metodologia COMBO (Ciclo Organizado Modular Baseado em Oportunidades) criada para permitir o desenvolvimento de jogos educacionais construtivistas baseados nos princípios de expressividade e explorabilidade. A metodologia foi avaliada e testada por meio do desenvolvimento do jogo Star Owners, que aborda alguns conceitos específicos, mas que não necessitam de conhecimento prévio do estudante/jogador. A elaboração da metodologia propõe mitigar lacunas e erros observados em iniciativas de aplicação e desenvolvimento de jogos educacionais e mostrou-se bem sucedida uma vez que foi realizado o desenvolvimento de um jogo fazendo o uso da COMBO e aplicando o mesmo em sala de aula.*

**Palavras-chaves.** *Serious Games; Construtivismo; Jogos na Educação; Metodologia.*

## 1. Introdução

As constantes revoluções no meio digital ao longo do último século têm gerado mudanças significativas no meio educacional [Collins and Halverson 2018], pois uma educação tecnológica eficiente não pode se encontrar deslocada das mudanças que o mundo vem vivendo [Malliarakis et al. 2013].

Nesse sentido aparecem os *serious games* como uma forma de usar jogos para ensino ou capacitação [Stokes 2005]. Seu uso tem ganhado popularidade no ensino de Computação [Bai et al. 2020] e tem sido bem aceito tanto por parte dos alunos quanto dos tutores [Dörner and Spierling 2014], apresentando uma dinâmica bem aceita inclusive por alunos que não se sentem intrinsecamente motivados por jogos [Hakulinen 2011].

A metodologia COMBO (Ciclo Organizado Modular Baseado em Oportunidades), utilizou-se de fundamentos do construtivismo, defendendo o aprender como um processo ativo [Wu et al. 2012] e fruto de interações passadas, permitindo a construção do conhecimento. Sendo assim, as etapas descritas nessa metodologia visam seguir dois princípios fundamentais de *serious games* construtivistas: expressividade e explorabilidade [Vahldick et al. 2016].

Com base nas instruções dessa metodologia, o jogo *Star Owners* foi desenvolvido com o objetivo de ensinar conceitos de Teoria da Computação para alunos de cursos superiores. O jogo representa um exemplo das possibilidades oferecidas por essa metodologia e é apresentado neste artigo, fornecendo um passo-a-passo para projetos futuros.

Esse trabalho identifica no uso de *serious games* construtivistas a possibilidade de ganhos no ensino superior de Computação e apresenta uma metodologia de desenvolvimento de jogos (COMBO) para esse fim.

Na seção 2 são discutidos os fundamentos teóricos do construtivismo, que embasam a metodologia COMBO, bem como é detalhada a COMBO, apresentando em etapas que a constituem. Na seção 3, o jogo *Star Owners* é apresentado. Por fim, a seção 4 apresenta os resultados obtidos, as conclusões e trabalhos futuros.

## **2. COMBO: uma metodologia para desenvolvimento de *serious games***

O ensino de Computação enfrenta desafios como a baixa interação entre alunos e professores, métodos de ensino inadequados e falta de interesse dos alunos [Kazimoglu et al. 2012]. Para enfrentar esses desafios, algumas abordagens propõem o uso de ferramentas de manipulação e visualização como parte da solução para o ensino de conteúdos abstratos em Computação, buscando conectar o mundo concreto à atmosfera da programação, proporcionando uma experiência mais prática e envolvente [Hsu and Wang 2018], podendo assim serem apresentadas na forma de um *serious game*.

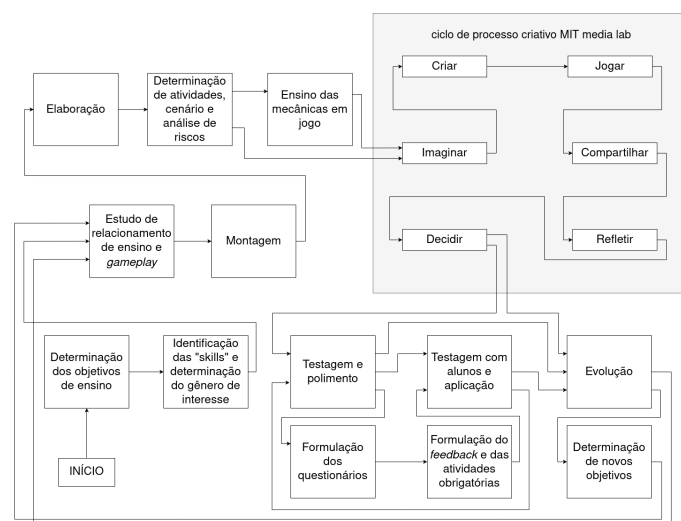
Jogos têm sido amplamente utilizados como uma ferramenta para minimizar a distância entre a teoria ensinada em sala de aula e a prática real na área de Computação. Por meio dos jogos, pode-se experimentar situações e desafios comuns da área, de forma prática e lúdica, ajudando a compreender e aplicar conceitos teóricos de forma mais efetiva [De Almeida Souza et al. 2017].

Mayer [Mayer 2012] identifica problemas comuns nos modelos para desenvolvimento de *serious games*, como: (1) pouca indicação de uso, (2) vago procedimento de validação, (3) problemas na hipótese, (4) falta de relações entre modelos e conceitos, (5) pouca operacionalização, (6) casos de estudos muito restritos, e (7) enfoque infantil. Além disso, o documento aponta para erros no desenvolvimento, como: (1) a falta de teorias, (2) de operacionalidade, (3) de validação (4) de um design de pesquisa, e (5) de ferramentas genéricas.

Desse modo, é possível avaliar um modelo pelo seu escopo, comparatividade, padronização, especificidade, flexibilidade, expansibilidade, validade, desobstrução, agilidade em questão de tempo e propósito [Mayer 2012]. Essas foram as diretrizes seguidas para o desenvolvimento da metodologia COMBO que visa auxiliar a criação de *serious games* construtivista voltados ao ensino de Computação.

COMBO segue o fluxograma ilustrado na Figura 1 baseando-se em diversos ci-

culos que guiam o desenvolvimento do jogo seguindo diretrizes preconizadas por Mayer [Mayer 2012].



**Figura 1. Diagrama da metodologia COMBO.**

Fonte: o autor.

- **Determinação do objetivo de ensino:** Objetivo de ensino, para esse documento, refere-se aos tópicos que se deseja ensinar a partir de *serious games*.
- **Identificação das *skills* e determinação do gênero de interesse:** Etapa onde relaciona-se habilidades (ou *skills* segundo Wassila & Tahar [Wassila and Tahar 2012]) com os objetivos de ensino estabelecidos, de maneira a obter uma indicação de qual o gênero de interesse para o jogo a ser elaborado [Wassila and Tahar 2012].
- **Estudo de relacionamento de ensino e *gameplay*:** Para que todos os elementos de *gameplay* com proposta educativa estejam claros para todo os *stakeholders*, é definida uma tabela de cenários do jogo com os seguintes atributos:
  - **Cenário educacional:** identifica o conteúdo que se deseja incentivar o aprendizado, de maneira técnica e clara;
  - **Cenário do jogo:** identifica como o conteúdo se apresenta no jogo;
  - **Mecânica (Rótulo em maiúsculo):** rotula a mecânica para melhor controle. Quando o cenário do jogo trata de uma mecânica a implementar e a ser identificada pelo aluno pelo princípio da explorabilidade;
  - **Estratégia (rótulo em minúsculo):** rotula a estratégia para melhor controle quando o cenário do jogo trata de uma estratégia ou caminho que o jogador poderá implementar pelo princípio da expressividade.
  - **Identificação ou implementação:** marca se a identificação da mecânica ou implementação da estratégia poderia ser imediata, trivial ou complexa dentro do contexto do jogo. Isso deve ser feito de forma livre e adaptativa, mas que guie os desenvolvedores nos ciclos dessa metodologia.
  - **Incentivo:** trabalha o sistema de incentivos para o jogador identificar ou implementar a mecânica ou estratégia quando de interesse.
  - **Elaboração:** é preenchido em etapas posteriores.

- **Montagem:** Etapa que descreve o funcionamento do jogo, sendo necessário realizá-la antes de iniciado o processo criativo, para garantir dois princípios da metodologia: que nenhuma mecânica ou estratégia diretamente ligada aos objetivos de ensino sejam despercebidos pelos alunos e; que todo o jogo possa ser jogado independente do conhecimento prévio da matéria por parte do jogador.
- **Elaboração:** Etapa de contemplação do estado atual de desenvolvimento do jogo, podendo representar um *timestamp* ou um indicativo do que já foi e do que ainda será elaborado. Nessa etapa é elaborada uma tabela de imprevistos e comportamentos indesejáveis que são observados ao longo de testes e/ou *feedbacks* obtidos.
- **Determinação de atividades, cenário e análise de riscos:** Etapa que representa a organização das atividades que seriam concluídas antes do começo do próximo ciclo, de forma a indicar para todos os desenvolvedores o que deve ser entregue, bem como os riscos até essa entrega.
- **Ensino das mecânicas do jogo:** Etapa a ser executada apenas após o formato do jogo estar mais claro, que poderá exigir alguns ciclos. Se dedica a indicar a necessidade de cuidado ao ensinar o jogador a jogar o jogo, para tanto, recomenda-se evitar conteúdos textuais e recordar o princípio do desconhecimento por parte do jogador [Kannappan et al. 2019]. O *feedback* do público alvo ou de testadores também é importante para essa etapa, que evolui conforme a necessidade do jogo.
- **Ciclo de processo criativo MIT Media Lab:** passo modular, que pode ser substituído por outros ciclos de processo criativo de acordo com a decisão da equipe de desenvolvimento. Representa efetivamente o processo de programação do jogo.
- **Testagem e polimento:** Salienta a necessidade de um preparo final antes da apresentação do jogo aos alunos e logo na saída do processo criativo, testes precisam ser aplicados, *bugs* consertados, códigos otimizados e mecânicas polidas.
- **Formulação dos questionários:** Visa a elaboração e o planejamento de distribuição de questionários que possam garantir que os alunos, de fato, jogaram o jogo. O questionário tem um caráter provisório e não cobraria do aluno conhecimentos sobre a disciplinas, sendo feito de perguntas simples.
- **Formulação do *feedback* e das atividades obrigatórias:** Visa identificar o que se espera que o aluno entregue da atividade do jogo, refletindo quais os níveis ou missões dentro do jogo eles devem priorizar e o que se espera de retorno.
- **Testagem e aplicação com alunos:** A proposta desta atividade é buscar não apenas cumprir o objetivo do jogo, mas também obter *feedback* dos alunos que possam levar a melhorias significativas no próximo ciclo. Recomenda-se que a atividade seja também uma forma de avaliação para os alunos, uma vez que o jogo foi desenvolvido com o objetivo de ser um recurso para estudo e reforço do aprendizado.
- **Evolução:** O engajamento do jogo é exercitado, e para tal se propôs a avaliação por níveis. Os níveis foram adaptados da proposta de engajamento apontada por Aditya et al. [Aditya et al. 2019], levando em consideração, também, as prioridades para o alcance de cada nível. Assim, considera-se nessa metodologia que o nível 1 representa a base do desenvolvimento do jogo, já os níveis seguintes devem ser considerados conforme o jogo toma formas mais desenvolvidas com o decorrer de cada ciclo.
- **Determinação de novos objetivos:** Após a etapa da evolução e se os desenvolvedores observarem ser possível, a partir das diretrizes já levantadas adicionar ao

fim da fila de prioridades novos objetivos.

### 3. Star Owners - um *Serious Game* para o ensino de Teoria da Computação desenvolvido utilizando a COMBO

*Star Owners* é um *serious game* desenvolvido fazendo uso da metodologia COMBO com o objetivo de contemplar as habilidades de compreender conteúdos, procedimentos, estruturas, sistemas e de tomada de decisão. Ele foi projetado para dar suporte à disciplina de Teoria da Computação, uma disciplina considerada obrigatória em diversos cursos de computação, além de apresentar um alto grau de dificuldade de aprendizado para boa parte dos alunos.

O jogo foi programado com o motor gráfico *Unity*, disponível gratuitamente para *Windows*, *Linux* e *Mac* e, de alta popularidade em termos de desenvolvimento de jogos profissionais e para iniciantes. A música e os efeitos sonoros foram compostas com a ferramenta LMMS (*Linux MultiMedia Studio*), enquanto que as imagens foram desenvolvidas por meio da ferramenta GIMP (*GNU Image Manipulation Program*).

O jogo está disponível para *Windows*, *Linux*, *Mac*, *Ios* e *Android*, por meio do link <sup>1</sup>. Nível após nível, os jogadores tem a possibilidade de simular o comportamento de uma das três estruturas que são apresentadas ao longo da disciplina: autômatos finitos, autômatos de pilha e máquinas de Turing.

#### 3.1. *Gameplay*

O objetivo principal do jogador é montar um caminho que leve um meteoro, de estação em estação até a estação dourada, de maneira a garantir que todos os quadrados sejam consumidos antes de alcançá-la, conforme ilustrado na Figura 2.

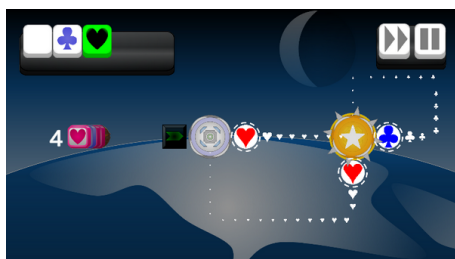


Figura 2. Tela que ilustra o jogo *Star Owners*.

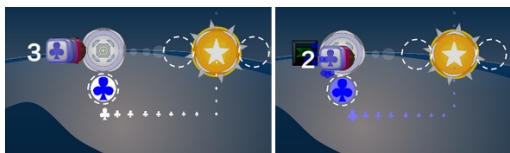
Fonte: o autor.

O comportamento dos meteoros, e seus quadrados, e a distribuição das estações imitam o conceito de autômatos (Figura 3). A configuração dos quadrados são equivalentes a uma entrada e as estações aos estados da máquina. Conforme os níveis ficam mais avançados, mais se exige da compreensão dessas mecânicas por parte do jogador, recompensando o mesmo por aprender cada vez mais os conceitos apresentados na disciplina.

Dessa forma, o princípio da metodologia COMBO é desenvolver jogos em que o aluno se sinta convidado a conciliar o que acabou de jogar com o que acabou de aprender.

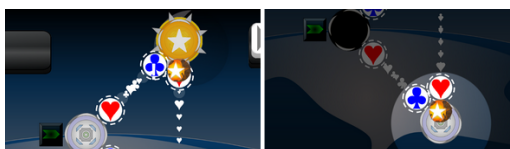
<sup>1</sup><https://drive.google.com/drive/folders/1bWWyOF4feOUZEgRjfvKuQlPYyboDjmVF>

Não havendo um caminho para o meteoro seguir com a configuração montada pelo jogador, a partida é encerrada. Se o jogador, no entanto, conseguir garantir que todos os meteoros cheguem a estação dourada, ele será recompensado com novos níveis mais desafiadores (Figura 4).



**Figura 3.** O símbolo presente no primeiro quadrado direciona o meteoro ao longo do caminho a seguir.

Fonte: o autor.



**Figura 4.** Caso o jogador não consiga cumprir a missão, a partida é encerrada.

Fonte: o autor.

### 3.2. Modos de jogo

No total, há três modos de jogo disponíveis ao jogador em cada fase:

- **Modo normal:** A construção da máquina é dinâmica e o jogador deverá adaptá-la para compensar cada entrada. O modo possui duas dificuldades, amador e profissional, onde na segunda, o jogador deverá lidar com o dobro de meteoros. Representa o modo principal do jogo e deverá ser concluído para liberar os dois modos de enigmas.
- **Modo enigma A:** Cinco meteoros são disponibilizados logo de início e o jogador precisará montar uma única máquina que aceite todas as entradas.
- **Modo enigma B:** A configuração é estática, cabendo ao jogador indicar cinco entradas que são aceitas pela máquina .

Os modos de enigma se aproximam mais do comportamento da máquina teórica, pois a adaptação dela passa a não ser mais dinâmica. O modo normal permite que os alunos treinem mais comportamentos e configurações possíveis, ao mesmo tempo que o desafio em tempo real recompensa o treinamento.

### 3.3. Mundos

Os níveis ao longo do jogo foram divididos em três mundos, contendo cinco fases, cada uma contemplando os três modos descritos.

Cada meteoro obtido com êxito aumenta o contador de meteoros que é necessário para que o jogador libere os novos mundos. Cada mundo liberado altera a jogabilidade de maneira a simular diferentes máquinas que são apresentadas ao longo da disciplina.

- **Mundo 1 - órbita da Terra:** Esse mundo foi desenvolvido para apresentar os conceitos de autômatos finitos, além de introduzir o jogo.

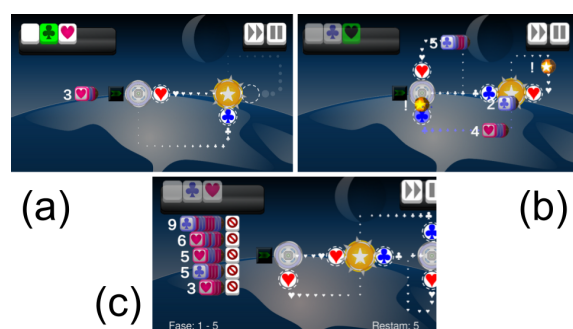


Figura 5. (a) Modo normal - jogador ajusta caminho para cada nova entrada; (b) Modo enigma A - jogador cria um percurso que aceite todas as sequências fornecidas; (c) Modo enigma B - jogador seleciona quais cadeias serão aceitas pelo caminho já definido.

Fonte: o autor.

- **Mundo 2 - cinturão de asteroides:** O aluno irá treinar os conceitos de autômatos de pilha. Dessa vez, o jogador deve se preocupar com as cores dos caminhos possíveis, sendo que branco significa que a pilha se encontra vazia e verde indica que há alguma carga.
- **Mundo 3 - o Sol:** Representa os últimos e mais desafiadores níveis, onde o jogador deverá montar máquinas de Turing que alterem as configurações dos quadros para os valores pedidos.

#### 4. Resultados Obtidos e Conclusão

O jogo *Star Owners* foi aplicado para uma turma de 47 alunos que cursaram a disciplina no primeiro semestre de 2023 na Universidade de São Paulo, campus de São Carlos. As atividades realizadas contaram como parte da nota da disciplina de Teoria de Computação.

Não sendo obrigatória a realização da atividade, nem todos os alunos realizaram todas as atividades propostas. A avaliação feita pelos alunos é ilustrada na Tabela 1, que contou com 36 respostas relacionadas à experiência pessoal. Além disso, na Tabela 2, respondida por 36 alunos, são ilustrados os critérios de engajamento propostos na escala de EGameflow [Fu et al. 2009]. As notas atribuídas pelos alunos foram dentro do intervalo de 1 a 5 para cada tópico.

Nos histogramas ilustrados na Figura 6 é possível observar a aplicação do jogo em uma turma de Teoria da computação em que foram computadas três ondas de aplicação do jogo. É possível observar na primeira coluna de histograma quantos alunos obtiveram quantas estrelas nos respectivos mundos. Na sequência, os histogramas ilustra o tempo gasto em minutos para a execução das tarefas, e na por fim as tentativas para concluir cada nível

Opiniões individuais e por escrito foram coletadas e também serão usadas para orientar os desenvolvedores nos próximos ciclos, com o objetivo de garantir melhorias na aplicação dessas atividades nas futuras turmas da disciplina.

A relação entre os cenários educacional e do jogo são ilustrados na Tabela 3. A metodologia COMBO permitiu a criação de um *serious game* (*Star Owners*) baseado nos princípios de expressividade e explorabilidade. Ela orientou todas as etapas do processo,



**Figura 6. Histograma com as informações de aplicação do jogo.**

Fonte: o autor.

**Tabela 1. Relação de notas atribuídas por alunos para cada critérios de engajamento propostos pelo EGameFlow [Fu et al. 2009].**

Tópico	1	2	3	4	5
Concentração	0	1	7	15	13
Clareza de Objetivos	1	5	10	13	7
Feedback	0	2	6	14	14
Desafio	0	4	6	14	12
Autonomia	0	4	6	11	15
Imersão	2	9	12	7	6
Interação social	4	8	7	11	6
Melhoria do conhecimento	1	1	9	12	13

Fonte: o autor.

**Tabela 2. Relação de notas atribuídas por alunos de acordo com suas próprias percepções da atividade.**

Tópico	1	2	3	4	5
Diversão	0	1	7	17	11
Intuitividade	0	4	11	3	18
Compreensão dos tópicos	0	2	6	17	11
Utilidade para com o aprendizado	0	3	12	16	5
Indicação futura da atividade	1	0	8	10	17

Fonte: o autor.

desde a concepção das ideias fundamentais até a aplicação do jogo no ensino superior de Computação. A abordagem começou do zero, incluindo a seleção do gênero e o engajamento do público-alvo.

O trabalho aqui apresentado colaborou para reduzir as fontes de frustração no aprendizado com uma metodologia, testada e avaliada por meio da proposição do jogo *Star Owners* que foi apreciado inclusive por estudantes que não possuíam conhecimento prévio da disciplina que o jogo aborda.

A utilização da COMBO permite a criação de jogos similares baseados em princípios semelhantes para outras disciplinas e tópicos. Essa abordagem pode ser aplicada em uma ampla gama de disciplinas desafiadoras e temas diversos, possibilitando o desenvolvimento de trabalhos futuros que explorem diferentes áreas de conhecimento. Além disso, como trabalho futuro tem-se ainda a realização de testes e aplicações do jogo *Star Owners* em novas turmas de graduação, dando sequência a novos ciclos da COMBO,



**Tabela 3. Tabela de cenários - cenário educacional e cenário do jogo, apresentados por meio dos ativos do *Star Owners*.**

Cenário educacional	Cenário do jogo	Rótulo	Identificação	Incentivo	Elaboração
Estado	Estações de redirecionamento, representados pelas formas circulares	a	Fácil	Nenhum	Feita
Transição	Caminhos associados de estações para estações	b	Fácil	Regra do jogo, fases iniciais instrutoras	Feita
Alfabeto terminal	O conjunto de símbolos (naipes)	c	Fácil	Nenhum	Feita
Estado inicial	Estação com uma seta grande	d	Imediata	Nenhum	Feita
Estado de aceitação	Estação dourada com uma estrela no centro	e	Necessária e não trivial	Pontuação, fases iniciais instrutoras	Feita
Aceitação	Alcançar o estado de aceitação com o símbolo da estrela	A	Necessária e não trivial	Pontuação, fases iniciais instrutoras	Feita
Aceitação inválida	Alcançar o estado de aceitação com o símbolo da bomba	B	Necessária	Penalização	Não feita
Rejeição	Não conseguir alcançar a aceitação	C	Necessária	Penalização	Feita
Determinismo	A cadeia não se divide	D	Não trivial	Nenhum	Feita
Símbolos terminais	Os naipes	f	Não trivial	Nenhum	Feita
Símbolos não Terminais	As estações	g	Difícil	Nenhum	Feita
Funcionamento real de um autômato	Níveis de enigmas	F	Fácil	Desafio bônus	Feita
<i>Push</i>	Se a cadeia passar pela meia lua verde clara, um elemento na pilha será acrescentado	i	Difícil	Design de níveis	Feita
<i>Pop</i>	Se a cadeia passar pela meia lua verde escura, um elemento da pilha será removido	j	Difícil	Design de níveis	Feita
Máquina de Turing, cabeça de leitura/escrita	Símbolo da cadeia destacado para cima	k	Difícil	Design de níveis	Feita
Máquina de Turing, mover para esquerda e direita	O jogador pode apertar na transição para alterar a direção que a cabeça (representado pelo símbolo destacado) se move	G	Difícil	Design de níveis	Feita
Máquina de Turing, leitura	A regra de leitura segue a mesma dinâmica das regras de transição	H	Imediata	Nenhum	Feita
Máquina de Turing, escrita	O jogador não controla a escrita, apenas a leitura. Quem estiver na cabeça, mudará de símbolo de acordo com o indicativo colorido da regra de transição	l	Difícil	Design de níveis	Feita
Máquina de Turing linearmente limitada	A cabeça não passa dos limites da própria entrada	m	Difícil	Nenhum	Feita
Máquina de Turing para operação	O jogador deverá montar um caminho para que a entrada iguale à saída	I	Difícil	Pontuação, apelo visual	Feita
Máquina de Turing para reconhecimento de linguagens	O jogador deverá montar um caminho para que a entrada alcance o estado de aceitação	J	Difícil	Pontuação, apelo visual	Feita
Gramática	No modo enigma A, o jogador deverá criar um autômato que aceite todas as cadeias indicadas	J	Difícil	Design de níveis	Feita
Linguagem	No modo enigma B, o jogador deverá identificar quais cadeias são aceitas pelo autômato montado	K	Difícil	Design de níveis	Feita
Autômatos finitos	O primeiro mundo do jogo foi baseado no funcionamento lógico de autômatos finitos	n	Fácil	Design de níveis	Feita
Autômatos de pilha	O segundo mundo do jogo foi baseado no funcionamento lógico de autômatos de pilha	o	Difícil	Design de níveis	Feita
Máquina de Turing	O terceiro mundo do jogo foi baseado no funcionamento lógico de máquinas de Turing	p	Difícil	Design de níveis	Feita

Fonte: o autor.

e também representando um meio avaliativo e possivelmente adaptativo da mesma.

## 5. Agradecimentos

Este estudo foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) e pela Comissão Gestora de Bolsas do PROEX-CCMC. Os levantamentos só foram possíveis graças ao auxílio do Prof. Dr. Thiago Alexandre Salgueiro Pardo e da participação de seus alunos voluntários.

## Referências

- Aditya, S. J., Santoso, H. B., and Isal, R. Y. K. (2019). Developing a game-based learning for branch and bound algorithm. In *2019 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 471–476.
- Bai, S., Hew, K. F., and Huang, B. (2020). Does gamification improve student learning outcome? evidence from a meta-analysis and synthesis of qualitative data in educational contexts. *Educational Research Review*, 30:100322.
- Collins, A. and Halverson, R. (2018). *Rethinking education in the age of technology: The digital revolution and schooling in America*. Teachers College Press.
- De Almeida Souza, M. R., Furtini Veado, L., Teles Moreira, R., Magno Lages Figueiredo, E., and Costa, H. A. X. (2017). Games for learning: bridging game-related education

- methods to software engineering knowledge areas. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)*, pages 170–179.
- Dörner, R. and Spierling, U. (2014). Serious games development as a vehicle for teaching entertainment technology and interdisciplinary teamwork: Perspectives and pitfalls. In *Proceedings of the 2014 ACM International Workshop on Serious Games, SeriousGames '14*, page 3–8, New York, NY, USA. Association for Computing Machinery.
- Fu, F.-L., Su, R.-C., and Yu, S.-C. (2009). Egameflow: A scale to measure learners' enjoyment of e-learning games. *Computers & Education*, 52:101–112.
- Hakulinen, L. (2011). Using serious games in computer science education. In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research, Koli Calling '11*, page 83–88, New York, NY, USA. Association for Computing Machinery.
- Hsu, C.-C. and Wang, T.-I. (2018). Applying game mechanics and student-generated questions to an online puzzle-based game learning system to promote algorithmic thinking skills. *Computers & Education*, 121:73–88.
- Kannappan, V. T., Fernando, O. N. N., Chattopadhyay, A., Tan, X., Hong, J. Y. J., Seah, H. S., and Lye, H. E. (2019). La petite fee cosmo: Learning data structures through game-based learning. In *2019 International Conference on Cyberworlds (CW)*, pages 207–210.
- Kazimoglu, C., Kiernan, M., Bacon, L., and MacKinnon, L. (2012). Learning programming at the computational thinking level via digital game-play. *Procedia Computer Science*, 9:522–531. Proceedings of the International Conference on Computational Science, ICCS 2012.
- Malliarakis, C., Satratzemi, M., and Xinogalos, S. (2013). A holistic framework for the development of an educational game aiming to teach computer programming. volume 1.
- Mayer, I. (2012). Towards a comprehensive methodology for the research and evaluation of serious games. *Procedia Computer Science*, 15:233–247. 4th International Conference on Games and Virtual Worlds for Serious Applications(VS-GAMES'12).
- Stokes, B. (2005). Videogames have changed: time to consider 'serious games'? *Development Education Journal*, 11:12.
- Vahldick, A., Mendes, A. J., and Marcelino, M. J. (2016). Towards a constructionist serious game engine. In *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016, CompSysTech '16*, page 361–368, New York, NY, USA. Association for Computing Machinery.
- Wassila, D. and Tahar, B. (2012). Using serious game to simplify algorithm learning. In *International Conference on Education and e-Learning Innovations*, pages 1–5.
- Wu, W.-H., Chiou, W.-B., Kao, H.-Y., Alex Hu, C.-H., and Huang, S.-H. (2012). Re-exploring game-assisted learning research: The perspective of learning theoretical bases. *Computers & Education*, 59(4):1153–1161.