



UNIVERSIDADE FEDERAL DO MARANHÃO - UFMA  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIAS - CCET  
COORDENAÇÃO DO CURSO DE ENGENHARIA DA COMPUTAÇÃO

**EMANUELLE VICTORIA FERNANDES SILVA**

**COMPILA: UMA JORNADA LÚDICA PELA  
ENGENHARIA DA COMPUTAÇÃO**

São Luís – MA

2026

**EMANUELLE VICTORIA FERNANDES SILVA**

**COMPILA: UMA JORNADA LÚDICA PELA  
ENGENHARIA DA COMPUTAÇÃO**

Trabalho de Conclusão de Curso apresentado para obtenção do título de Bacharel em Engenharia da Computação, pela Universidade Federal do Maranhão - UFMA, Cidade Universitária: Campus Dom Delgado, São Luís, MA.

Orientador: Prof. Dr. Alex Oliveira Barradas Filho

São Luís – MA

2026

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).  
Diretoria Integrada de Bibliotecas/UFMA

Fernandes Silva, Emanuelle Victoria.

COMPILA: UMA JORNADA LÚDICA PELA ENGENHARIA DA  
COMPUTAÇÃO / Emanuelle Victoria Fernandes Silva. - 2026.  
62 f.

Orientador(a): Alex Oliveira Barradas Filho.

Curso de Engenharia da Computação, Universidade Federal  
do Maranhão, São Luís, Maranhão, 2026.

1. Metodologias Ativas. 2. Aprendizagem Lúdica. 3.  
Jogos de Tabuleiro. 4. Ensino de Computação. 5.  
Engenharia da Computação. I. Barradas Filho, Alex  
Oliveira. II. Título.

**EMANUELLE VICTORIA FERNANDES SILVA**

**COMPILA: UMA JORNADA LÚDICA PELA  
ENGENHARIA DA COMPUTAÇÃO**

Aprovada em: 23 / 01/ 2026

BANCA EXAMINADORA

---

**Prof. Dr. Alex Oliveira Barradas Filho (Orientador)**

UFMA

---

**Prof. Dr. Bruno Feres de Souza**

UFMA

---

**Prof. Dr. Davi Viana dos Santos**

UFMA

São Luís – MA

2026

## AGRADECIMENTOS

Agradeço, primeiramente, a Deus, por me conceder força e sustento ao longo de toda essa trajetória. Sem Ele, nada disso teria sido possível.

Agradeço aos meus pais, Silvana Fernandes Silva e José de Arimatéa, pelo incentivo constante na busca pelo conhecimento e pelo apoio.

Ao meu irmão, Eliaquim Fernandes Silva, agradeço pelos momentos de descontração, que tornaram essa caminhada mais leve.

Ao meu namorado, Brenno Pacheco Carneiro, pelo apoio constante, pelo carinho e pelo incentivo incondicional, especialmente nos momentos mais desafiadores desta trajetória. Sua presença e compreensão foram fundamentais para que este trabalho se tornasse possível.

Ao Spike Lee, que, mesmo sem compreender prazos ou normas acadêmicas, soube estar presente nos momentos em que mais precisei, lembrando-me diariamente do valor do afeto e da constância.

Ao meu orientador, Prof. Dr. Alex Oliveira Barradas Filho, por mais uma vez contribuir de forma significativa para o meu desenvolvimento acadêmico e para a realização deste trabalho.

Aos meus amigos, por tornarem essa jornada mais leve e significativa.

Por fim, agradeço a todos que, direta ou indiretamente, contribuíram para a realização deste projeto. Meu sincero agradecimento pelo tempo, conhecimento e apoio compartilhados, fundamentais para a concretização deste trabalho.

“Para que todos vejam, e saibam, e considerem, e  
juntamente entendam que a mão do Senhor fez isto.”  
Isaías 41:2

## RESUMO

A evasão e a retenção no ensino superior constituem desafios persistentes, especialmente em cursos das áreas de Ciências Exatas, Engenharias e Computação, nos quais elevados índices de reprovação e dificuldades conceituais são frequentemente observados nos períodos iniciais da graduação. Nesse contexto, este trabalho tem como objetivo desenvolver um jogo de tabuleiro educativo, denominado COMPILA, como recurso pedagógico complementar voltado ao apoio do processo de ensino-aprendizagem em cursos de Computação, com foco na Engenharia da Computação da Universidade Federal do Maranhão (UFMA). A pesquisa caracteriza-se como aplicada, de abordagem qualitativa e caráter exploratório. O desenvolvimento do jogo fundamentou-se na análise dos Projetos Pedagógicos de Curso (PPC) e das ementas das disciplinas, organizando os conteúdos em fases formativas que representam, de forma lúdica, a trajetória acadêmica do estudante. O COMPILA incorpora princípios de metodologias ativas e aprendizagem lúdica, utilizando desafios progressivos, tomada de decisão e interação social como elementos centrais de sua dinâmica. Para avaliação do recurso desenvolvido, foi aplicado o questionário System Usability Scale (SUS) a seis estudantes da área de Computação, após a realização de uma partida do jogo. Os resultados indicaram uma pontuação média de 85,8 no SUS, valor considerado elevado segundo os critérios de interpretação do instrumento, sugerindo boa aceitação, clareza das regras e facilidade de uso do jogo. Embora não tenha como finalidade mensurar impactos diretos sobre aprendizagem ou redução da evasão e retenção, o estudo evidencia que o COMPILA apresenta condições adequadas de usabilidade que justificam sua adoção como ferramenta pedagógica complementar em contextos educacionais, especialmente em propostas fundamentadas em metodologias ativas e abordagens lúdicas no ensino de Computação.

**Palavras-chave:** Metodologias ativas. Aprendizagem lúdica. Jogos de tabuleiro. Ensino de Computação. Engenharia da Computação.

## ABSTRACT

Dropout and retention in higher education remain persistent challenges, especially in programs related to Exact Sciences, Engineering, and Computing, where high failure rates and conceptual difficulties are frequently observed in the early semesters of undergraduate studies. In this context, this study aims to develop an educational board game, called COMPILA, as a complementary pedagogical resource to support the teaching and learning process in Computing programs, with a focus on Computer Engineering at the Federal University of Maranhão (UFMA). The research is characterized as applied, with a qualitative approach and an exploratory nature. The development of the game was based on the analysis of the Course Pedagogical Projects (PPC) and course syllabi, organizing the curricular contents into formative phases that ludically represent the students' academic trajectory. COMPILA incorporates principles of active learning methodologies and playful learning, using progressive challenges, decision-making, and social interaction as central elements of its dynamics. To evaluate the developed resource, the System Usability Scale (SUS) questionnaire was applied to six Computing students after a game session. The results indicated an average SUS score of 85.8, which is considered high according to the instrument's interpretation criteria, suggesting good acceptance, clear rules, and ease of use. Although the study does not aim to measure direct impacts on learning outcomes or reductions in dropout and retention rates, the findings indicate that COMPILA presents adequate usability conditions to justify its adoption as a complementary pedagogical tool in educational contexts, particularly within proposals grounded in active methodologies and playful approaches to Computing education.

**Keywords:** Active learning methodologies. Playful learning. Board games. Computing education. Computer Engineering.

## LISTA DE ILUSTRAÇÕES

<b>Figura 1</b> – Fluxograma do processo metodológico de desenvolvimento e avaliação do jogo COMPILA .....	26
<b>Figura 2</b> - Representação ilustrativa das fases do jogo COMPILA a partir da análise dos Projetos Pedagógicos de Curso do BICT e da Engenharia da Computação.....	28
<b>Figura 3</b> – Fluxo metodológico para elaboração das questões e cartas do jogo COMPILA...	31
<b>Figura 4</b> – Tabuleiro do jogo COMPILA .....	35
<b>Figura 5</b> – Frente das cartas do jogo COMPILA.....	36
<b>Figura 6</b> – Verso das cartas do jogo COMPILA.....	37
<b>Figura 7</b> – Personagens do jogo COMPILA .....	38

## LISTA DE ABREVIATURAS E SIGLAS

ABJ	Aprendizagem Baseada em Jogos
ABP	Aprendizagem Baseada em Problemas
BICT	Bacharelado Interdisciplinar em Ciência e Tecnologia
BNCC	Base Nacional Comum Curricular
IES	Instituição de Ensino Superior
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
MA	Metodologias Ativas de Aprendizagem
MEC	Ministério da Educação
PPC	Projeto Pedagógico de Curso
REUNI	Programa de Apoio a Planos de Reestruturação e Expansão das Universidades Federais
SBC	Sociedade Brasileira de Computação
SUS	System Usability Scale
TSG	Taxa de Sucesso na Graduação
UFMA	Universidade Federal do Maranhão

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>12</b>
<b>1.1</b>	<b>Evasão e Retenção no Ensino Superior em Cursos da Computação .....</b>	<b>13</b>
1.1.1	Conceitos e dimensões de evasão e retenção no ensino superior .....	13
1.1.2	Determinantes e contexto institucional: expansão, REUNI e fatores associados.....	14
1.1.3	Evasão e retenção na Computação: gargalos, reprovação e dificuldades conceituais.....	16
<b>1.2.</b>	<b>Aspectos do Processo de Aprendizagem em Disciplinas de Computação .....</b>	<b>17</b>
<b>1.3.</b>	<b>Importância de Metodologias Ativas .....</b>	<b>19</b>
<b>1.4</b>	<b>Estado da arte .....</b>	<b>23</b>
<b>2</b>	<b>OBJETIVOS .....</b>	<b>25</b>
<b>2.1</b>	<b>Objetivo Geral .....</b>	<b>25</b>
<b>2.2</b>	<b>Objetivos Específicos.....</b>	<b>25</b>
<b>3</b>	<b>METODOLOGIA.....</b>	<b>26</b>
<b>3.1</b>	<b>Tipo e abordagem da pesquisa .....</b>	<b>26</b>
<b>3.2</b>	<b>Contexto da pesquisa e concepção do jogo COMPILA .....</b>	<b>27</b>
<b>3.3</b>	<b>Procedimentos de Avaliação do Jogo COMPILA .....</b>	<b>32</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÕES .....</b>	<b>35</b>
<b>5</b>	<b>CONCLUSÃO.....</b>	<b>40</b>
	<b>REFERÊNCIAS .....</b>	<b>42</b>
	<b>APÊNDICE A – TABELA DAS QUESTÕES UTILIZADAS NO JOGO.....</b>	<b>51</b>
	<b>APÊNDICE B – TABELA DOS PERSONAGENS E SUAS HABILIDADES .....</b>	<b>60</b>
	<b>APÊNDICE C – QUESTIONÁRIO DE AVALIAÇÃO DO JOGO .....</b>	<b>61</b>
	<b>APÊNDICE D – TABELA DE RESPOSTAS INDIVIDUAIS DE AVALIAÇÃO .....</b>	<b>63</b>

## 1 INTRODUÇÃO

A expansão do ensino superior brasileiro, intensificada a partir das primeiras décadas do século XXI, trouxe avanços significativos no que se refere ao acesso à universidade e à democratização das oportunidades educacionais. Políticas públicas voltadas à ampliação de vagas, à interiorização das universidades federais e à implementação de ações afirmativas contribuíram para modificar o perfil do estudante universitário, incorporando grupos historicamente excluídos do sistema educacional. Contudo, esse movimento de expansão não foi acompanhado, na mesma proporção, por políticas estruturais e pedagógicas eficazes voltadas à permanência e à conclusão dos cursos de graduação, resultando na persistência e, em alguns casos, no agravamento dos fenômenos da evasão e da retenção no ensino superior (Paula, 2017; Araujo; Mariano; Oliveira, 2021).

Nesse contexto, a evasão e a retenção passaram a ser compreendidas não apenas como problemas individuais, atribuídos às dificuldades pessoais do estudante, mas como fenômenos complexos e multifatoriais, fortemente associados às condições institucionais, pedagógicas, socioeconômicas e curriculares das instituições de ensino superior (Hoed, 2016; Máximo et al., 2024). A não conclusão de um curso superior representa perdas significativas em diferentes dimensões: para o estudante, que vivencia frustrações acadêmicas e profissionais; para a instituição, que enfrenta ineficiência no uso de recursos humanos e financeiros; e para a sociedade, que deixa de contar com profissionais qualificados em áreas estratégicas para o desenvolvimento econômico e tecnológico (Saccaro; França; Jacinto, 2019).

No Brasil, dados recentes evidenciam a gravidade desse cenário. Entre os anos de 2018 e 2022, a taxa de evasão no ensino superior atingiu aproximadamente 57,2%, segundo o Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). Tal índice revela que mais da metade dos estudantes que ingressam em cursos de graduação não alcançam a conclusão de sua formação, mesmo diante do elevado investimento público destinado à educação superior. Estima-se que, em 2015, o custo anual médio de manutenção de um estudante em uma universidade pública federal tenha sido superior a R\$ 23.000,00, valor que evidencia o impacto financeiro associado à retenção prolongada e à evasão definitiva (Araujo; Mariano; Oliveira, 2021).

Embora a evasão e a retenção sejam fenômenos observados em diversas áreas do conhecimento, a literatura aponta que cursos das áreas de Ciências Exatas, Engenharias e Computação apresentam índices significativamente mais elevados quando comparados a áreas como Direito, Saúde ou Ciências Sociais Aplicadas (Saccaro; França; Jacinto, 2019). No campo

da Computação, em particular, esses fenômenos assumem contornos ainda mais críticos, em razão das características específicas da formação, que exige do estudante elevado nível de abstração, raciocínio lógico, capacidade de resolução de problemas complexos e domínio de fundamentos matemáticos e algorítmicos.

Diante desse cenário, emerge a necessidade de investigar alternativas pedagógicas capazes de contribuir para a melhoria do processo de ensino-aprendizagem em cursos da área de Computação, especialmente nos períodos iniciais da graduação, nos quais se concentram os maiores índices de reprovação, retenção e evasão. Entre essas alternativas, destacam-se as metodologias ativas de aprendizagem e o uso de abordagens lúdicas, que têm sido apontadas na literatura como estratégias promissoras para promover maior engajamento discente, aprendizagem significativa e permanência acadêmica.

É nesse contexto que se insere o presente trabalho, que propõe o desenvolvimento de um jogo de tabuleiro educativo intitulado COMPILA, voltado à representação lúdica da trajetória acadêmica em cursos de Computação. O nome do jogo faz referência ao processo de compilação, fundamental na área da Computação, no qual diferentes etapas e instruções são organizadas e transformadas em um produto executável. De forma análoga, o jogo busca representar a construção progressiva do conhecimento ao longo da graduação, na qual os conteúdos adquiridos em diferentes momentos do curso são integrados até a conclusão da formação acadêmica. Assim, o COMPILA pretende contribuir, de maneira lúdica e cooperativa, para a reflexão sobre o percurso formativo e os desafios enfrentados pelos estudantes ao longo da graduação em Engenharia da Computação.

A proposição de um recurso pedagógico com essa finalidade exige, contudo, a compreensão dos fenômenos educacionais que motivam sua concepção. Nesse sentido, torna-se fundamental discutir os conceitos de evasão e retenção no ensino superior, especialmente em cursos da área da Computação, uma vez que esses fenômenos constituem o plano de fundo que justifica a adoção de estratégias pedagógicas alternativas, como o uso de abordagens lúdicas e metodologias ativas.

## **1.1 Evasão e Retenção no Ensino Superior em Cursos da Computação**

### **1.1.1 Conceitos e dimensões de evasão e retenção no ensino superior**

A análise do fluxo acadêmico no ensino superior exige precisão conceitual, pois termos como evasão e retenção orientam diagnósticos institucionais e a formulação de políticas educacionais (Hoed, 2016). A evasão é definida como a “saída definitiva do aluno de seu curso de origem, sem concluí-lo” (MEC, 1996, p. 15). Em termos analíticos, trata-se da interrupção

do ciclo de estudos por motivos distintos da diplomação, podendo ocorrer em diferentes níveis: evasão de curso (desligamento da graduação específica), evasão da instituição (desligamento da IES) e evasão do sistema (abandono do ensino superior) (Hoed, 2016). Nessa discussão, o INEP diferencia evasão de abandono: enquanto a evasão implica saída definitiva, o abandono corresponde à interrupção temporária, com possibilidade de retorno posterior, além de haver classificações que distinguem evasão temporária e definitiva conforme a expectativa de reingresso (Hoed, 2016).

Em contraste, a retenção refere-se ao prolongamento da permanência do estudante na instituição além do tempo regular previsto para integralização curricular (Melo; Naves, 2015). No Brasil, o termo tende a assumir conotação negativa, associada à ineficiência do fluxo acadêmico, enquanto em abordagens internacionais o conceito de *retention* é por vezes relacionado à permanência até que o estudante alcance seus objetivos formativos (Santos, 2015). É importante notar que evasão e retenção são fenômenos distintos, porém interligados: o trancamento de matrícula, por exemplo, não caracteriza evasão, mas pode ampliar o tempo de curso e, conseqüentemente, intensificar a retenção (Campello; Lins, 2008). De modo geral, a literatura aponta a reprovação sucessiva como um dos principais mecanismos que prolongam a trajetória acadêmica e aumentam custos institucionais, podendo também fragilizar o vínculo do estudante com o curso e elevar o risco de desligamento definitivo (Hoed, 2016; Melo; Naves, 2015). Assim, compreender esses conceitos é essencial, pois elevados índices de evasão e retenção produzem desperdício de recursos e impactos negativos sobre a trajetória acadêmica e profissional do discente (Teodoro; Kappel, 2020; Cunha; Nascimento; Durso, 2014).

### 1.1.2 Determinantes e contexto institucional: expansão, REUNI e fatores associados

A compreensão da evasão e da retenção no ensino superior brasileiro está fortemente ligada às transformações institucionais e às políticas públicas de expansão do acesso, intensificadas a partir dos anos 2000 (Paula, 2017). Nesse cenário, destaca-se o REUNI, instituído pelo Decreto nº 6.096/2007, como política voltada à ampliação de vagas, reestruturação curricular e ocupação de vagas ociosas, articulando a expansão da oferta à expectativa de reduzir evasão e promover permanência (Predes; Junior, 2014; Sousa; Nascimento, 2018). Os efeitos do programa incluem crescimento expressivo das matrículas em cursos presenciais entre 2006 e 2014 e mudanças relevantes no perfil discente, com maior

participação de estudantes oriundos de escolas públicas e de contextos socioeconômicos mais vulneráveis (Saccaro; França; Jacinto, 2019; Lima, 2024).

Entretanto, a literatura aponta que a expansão do acesso não foi acompanhada com a mesma intensidade por condições estruturais e pedagógicas capazes de sustentar permanência e conclusão. Nesse sentido, observa-se queda em indicadores de conclusão e diplomação e redução da Taxa de Sucesso na Graduação (TSG), fenômeno associado ao aumento de retenção e evasão no período subsequente à implementação do REUNI (Paula, 2017; Lima, 2024; Silva et al., 2017). Em paralelo, o PNAES foi instituído para oferecer condições materiais mínimas de permanência aos estudantes em vulnerabilidade socioeconômica (Brasil, 2007; Lima, 2024). Ainda assim, autores discutem a permanência de barreiras estruturais e pedagógicas, apontando o fenômeno de “inclusão excludente”, no qual o estudante consegue acessar o ensino superior, mas enfrenta dificuldades que comprometem seu êxito acadêmico e favorecem o abandono (Paula, 2017).

Além do contexto institucional, a evasão e a retenção resultam de um conjunto multifatorial de determinantes. Parte da literatura organiza as causas em fatores individuais, institucionais e externos (Silva et al., 2017) ou em dimensões econômicas, vocacionais e institucionais (Barroso; Falcão, 2004). Entre os fatores individuais, destaca-se a condição socioeconômica, que pode exigir conciliação entre trabalho e estudo e comprometer desempenho e permanência, ao passo que auxílios e apoio financeiro se relacionam a melhores indicadores de retenção (Gaioso, 2005; Garcia; Gomes, 2022; Saccaro; França; Jacinto, 2019). Fragilidades na formação básica, especialmente em Matemática, também se refletem nas disciplinas iniciais e afetam o aproveitamento acadêmico, elevando risco de retenção e evasão (Santos, 2022; Barcelos; Silveira, 2012; Junior et al., 2023).

No plano institucional, práticas pedagógicas tradicionais e pouco contextualizadas, currículos extensos com pré-requisitos rígidos e critérios avaliativos inadequados são frequentemente associados a dificuldades de aprendizagem e desmotivação (Adachi, 2009; Hoed, 2016; Garcia; Gomes, 2022). Soma-se a isso a dimensão vocacional, relacionada à falta de afinidade com o curso e à discrepância entre expectativas e realidade acadêmica (Barroso; Falcão, 2019; Hoed, 2016). Por fim, a saúde mental tem se consolidado como fator relevante, com evidências de que ansiedade e sofrimento psíquico afetam desempenho e permanência em IFES (Santos, 2022; Junior et al., 2023).

### 1.1.3 Evasão e retenção na Computação: gargalos, reprovação e dificuldades conceituais

Embora a evasão e a retenção ocorram em diferentes áreas, cursos de Computação apresentam historicamente índices elevados, constituindo desafio persistente em âmbito nacional e internacional (Hoed, 2016). Em contexto internacional, estudos apontam taxas relevantes já no primeiro ano, com diferenças marcantes em relação a cursos como Medicina e Direito (Mooney et al., 2010). No Brasil, análises indicam taxas expressivas em áreas de Ciências, Matemática, Engenharia e Computação, com destaque para a magnitude do problema em comparação a áreas com maior estabilidade (Saccaro; França; Jacinto, 2019), além de estimativas preocupantes no setor privado (SEMESP, 2012).

A literatura converge ao apontar as disciplinas introdutórias, especialmente as de programação, como gargalos do percurso formativo, frequentemente associadas a elevados índices de reprovação (Junior, 2009; Silva; Falcão, 2019). O ensino de programação envolve desafios específicos: demanda raciocínio lógico e algorítmico, capacidade de resolução de problemas e domínio de abstrações, além da necessidade de expressar soluções com sintaxe formal de linguagens de programação (Hoed, 2016). Revisões indicam que, na perspectiva discente, dificuldades recorrentes envolvem resolução de problemas, abstração e raciocínio lógico, enquanto docentes destacam raciocínio lógico, abstração e resolução de problemas como competências essenciais para o sucesso na aprendizagem (Hoed, 2016; Silva; Falcão, 2019).

A reprovação em disciplinas consideradas críticas, como Algoritmos e Cálculo, tem sido apontada como fator de risco significativo, sobretudo por ocorrer nos períodos iniciais, quando se observam picos de evasão e maior vulnerabilidade acadêmica (Santos, 2015; Hoed, 2016). A retenção tende a emergir como efeito cumulativo dessas reprovações, prolongando a permanência do estudante, elevando custos institucionais e, simultaneamente, fragilizando o vínculo com o curso, o que pode culminar em evasão (Araujo; Mariano; Oliveira, 202; Santos, 2015). Além disso, fatores vocacionais e informacionais também são relevantes, pois muitos estudantes ingressam com expectativas distorcidas sobre o que é um curso de Computação, o que pode gerar frustração e abandono precoce (Hoed, 2016). Nesse debate, o Pensamento Computacional aparece como abordagem potencial para desenvolver competências como abstração, decomposição e avaliação de soluções, embora sua integração efetiva ao ensino superior e seus efeitos na redução de retenção e evasão ainda sejam discutidos na literatura e nos referenciais da área (SBC, 2017; Silva; Falcão, 2019).

## 1.2. Aspectos do Processo de Aprendizagem em Disciplinas de Computação

O processo de aprendizagem em disciplinas de Computação, especialmente aquelas relacionadas à programação, é amplamente reconhecido como desafiador, sobretudo nos semestres iniciais da graduação. Isso ocorre porque a área impõe elevadas exigências cognitivas, demandando do estudante a integração entre conceitos teóricos, raciocínio lógico e fundamentos matemáticos, além da capacidade de transformar problemas em soluções estruturadas e passíveis de implementação (Brito, 2019; Bosse, 2020). Nesse sentido, disciplinas introdutórias como Algoritmos e Lógica de Programação frequentemente apresentam altos índices de reprovação, o que evidencia a complexidade envolvida na construção do pensamento computacional e na aprendizagem de estratégias eficazes de resolução de problemas (Brito, 2019; Arimoto; Oliveira, 2019; Shimasak; Prado, 2021).

De modo geral, a literatura aponta que parte dessas dificuldades se relaciona à própria natureza abstrata dos conteúdos de Computação. Programar exige, antes de codificar, formular modelos mentais e representações abstratas que permitam compreender e estruturar o problema em termos computacionais (Franzen, 2019). A abstração, nesse cenário, é frequentemente citada como um indicador de sucesso na aprendizagem de programação, pois envolve identificar padrões, extrair características essenciais, generalizar comportamentos e definir funcionalidades antes da implementação concreta (Franzen, 2019; Neto, 2025). Em Programação Orientada a Objetos (POO), por exemplo, abstrair implica selecionar atributos relevantes de entidades do mundo real e convertê-los em estruturas como classes e objetos, ignorando detalhes secundários. Entretanto, conceitos como encapsulamento, herança e polimorfismo, por serem intrinsecamente abstratos, tornam-se particularmente difíceis de compreender e aplicar para estudantes iniciantes (Neto, 2025).

Além disso, a abstração está associada ao próprio Pensamento Computacional, que pressupõe raciocínio abstrato e capacidade de generalização como competências centrais (Boucinha, 2017). A transição do pensamento cotidiano, frequentemente mais “concreto”, para o pensamento “computacional” constitui, portanto, uma barreira importante: muitos estudantes relatam dificuldade em converter problemas que seriam resolvidos manualmente em algoritmos claros e determinísticos, adequados ao funcionamento das máquinas (Bosse, 2020). Essa mudança exige operar com lógica sequencial, precisão e formalização, características que nem sempre foram suficientemente desenvolvidas na educação básica, onde o raciocínio lógico pode ter sido pouco estimulado (Nunes, 2023).

Esse desafio se intensifica porque a Computação se sustenta em bases lógico-matemáticas rigorosas, como a lógica formal e a teoria dos conjuntos, o que reforça a necessidade de domínio conceitual para a formulação de algoritmos coerentes (Neto, 2009; Brito, 2019). A lógica, entendida como o estudo dos princípios que distinguem raciocínios válidos de inválidos, é essencial para organizar o pensamento e estruturar soluções computacionais com consistência (Nunes, 2023). Assim, não é surpresa que o desenvolvimento do raciocínio lógico seja frequentemente apresentado como competência-chave no aprendizado de programação, pois programar envolve construir soluções baseadas em regras formais e relações lógicas (Rapkiewicz, 2006; Rapkiewicz et al., 2006; Martins; Marin; Alves, 2024).

Dentro desse processo, o pensamento algorítmico emerge como elemento estruturante, sendo compreendido como a capacidade de formular uma sequência ordenada e finita de passos para resolver um problema (Silva, 2019). Em disciplinas de Algoritmos, o objetivo central consiste justamente em desenvolver a habilidade de analisar problemas e expressar suas soluções de forma sistemática por meio de algoritmos (Rapkiewicz, 2006). Entretanto, diversos estudantes enfrentam dificuldades em dominar estruturas fundamentais de controle, como sequenciação, decisão e repetição, as quais dependem de operadores lógicos e relacionais que determinam o fluxo de execução do programa (Silva, 2019; Martins; Marin; Alves, 2024). Nesses casos, o problema não se restringe à sintaxe: a dificuldade recai sobretudo sobre a compreensão da semântica e do comportamento do algoritmo durante a execução (Bosse, 2020). Quando esse domínio não é alcançado, observa-se limitação na construção de soluções consistentes e aumento da ocorrência de erros persistentes de natureza lógica e semântica (Franzen, 2019; Bosse, 2020).

Outro aspecto decisivo envolve a dificuldade do estudante em lidar com estruturas “invisíveis” durante a execução de um programa, como fluxo de controle, estados internos, pilha de chamadas em processos recursivos e organização da memória. Para iniciantes, visualizar mentalmente essas estruturas abstratas é um desafio significativo, pois exige a construção de representações internas sobre processos que não são observáveis diretamente (Santiago; Kronbauer, 2017). Mesmo algoritmos considerados simples podem gerar hierarquias complexas de execução, o que compromete a compreensão global do funcionamento do programa. Nessas situações, a atenção excessiva à sintaxe da linguagem, que deveria ser apenas um meio, tende a desviar o foco do objetivo principal: a elaboração do raciocínio e da solução algorítmica (Bosse, 2020; Santiago; Kronbauer, 2017).

Somado a isso, a literatura critica o distanciamento entre teoria e prática no ensino de Computação, especialmente quando predominam metodologias tradicionais baseadas na lógica

“teoria–exercício”. Nesse modelo, o ensino se concentra na exposição de conteúdos e na demonstração de aspectos sintáticos das linguagens, produzindo uma aprendizagem frequentemente mecânica e pouco significativa (Silva, 2019; Rapkiewicz et al., 2006). O uso recorrente de exercícios artificiais, descontextualizados e com baixa conexão com situações reais contribui para a desmotivação discente, tornando o processo monótono e dificultando que o estudante compreenda a aplicabilidade dos conceitos aprendidos (Rapkiewicz et al., 2006; Franzen, 2019; Neto, 2025). Ademais, quando os conceitos são ensinados de forma isolada, sem conexões com problemas autênticos, há prejuízo na retenção do conhecimento e na capacidade de transferir o que foi aprendido para novos contextos (Rapkiewicz et al., 2006; Santiago; Kronbauer, 2017).

Essa fragilidade pedagógica também se manifesta na forma como a avaliação é conduzida. Em muitos casos, a avaliação assume caráter classificatório, orientado para a reprodução de conteúdo e para a verificação do domínio sintático, o que limita o desenvolvimento de competências como análise, criatividade, experimentação e resolução de problemas (Nunes, 2023; Cintra, 2023). Como resultado, o estudante tende a operar com memorização mecânica e a evitar o erro, em vez de utilizá-lo como parte constitutiva do processo de aprendizagem, o que é particularmente problemático em uma área na qual testar hipóteses, depurar falhas e iterar soluções constituem práticas centrais (Rapkiewicz et al., 2006).

À luz desses aspectos, é possível compreender que a aprendizagem em Computação não se resume ao domínio de uma linguagem de programação. Trata-se de um processo que exige a articulação de abstração, formalismo lógico-matemático, pensamento algorítmico e capacidade de representar mentalmente estruturas invisíveis de controle e memória (Nunes, 2023; Brito, 2019; Santiago; Kronbauer, 2017). Quando essas habilidades não são fortalecidas, sobretudo nos semestres iniciais, o estudante tende a vivenciar dificuldades recorrentes, baixo desempenho acadêmico e desmotivação, elementos frequentemente associados ao aumento de reprovação, retenção e, em casos mais críticos, à evasão (Shimasak; Prado, 2021; Martins; Marin; Alves, 2024).

### **1.3. Importância de Metodologias Ativas**

As transformações recentes no cenário educacional, especialmente no ensino superior, têm intensificado o debate sobre a necessidade de práticas pedagógicas mais alinhadas às demandas contemporâneas de formação. Em cursos da área de Computação, tal necessidade se

torna ainda mais evidente, uma vez que a aprendizagem exige autonomia, pensamento crítico, capacidade de resolução de problemas e desenvolvimento de competências socioemocionais, além do domínio técnico (Calderon; Silva; Feitosa, 2021; Santos et al., 2025). Nesse contexto, as Metodologias Ativas de Aprendizagem (MAs) consolidam-se como uma alternativa relevante ao modelo tradicional, ao reposicionarem o estudante como protagonista do processo educativo e promoverem maior engajamento, participação e construção significativa do conhecimento (Filho, 2024; Schwengber, 2024).

De modo geral, as metodologias ativas podem ser compreendidas como estratégias pedagógicas centradas na participação efetiva do estudante na construção do conhecimento. Diferentemente das abordagens tradicionais, nas quais o professor ocupa posição central como transmissor do conteúdo e o aluno assume papel predominantemente passivo, as MAs valorizam a experiência prévia do discente, a colaboração e a investigação, favorecendo aprendizagens mais contextualizadas e com maior sentido para o estudante (Schwengber, 2024; Dias; Araújo, 2024). Assim, o processo educativo passa a privilegiar práticas construtivas e desafiadoras, estimulando capacidades críticas e reflexivas por meio de experiências que articulam teoria e prática (Dias; Araújo, 2024; Calderon; Silva; Feitosa, 2021).

Essa mudança de paradigma representa uma ruptura com o ensino tradicional baseado em aulas expositivas extensas. Embora esse modelo possa ser eficiente para apresentar conteúdos teóricos a grandes grupos, estudos indicam que ele tende a ser desestimulante, favorecendo dispersão e perda de atenção em curtos intervalos, além de reforçar a reprodução mecânica de conteúdos em detrimento da construção ativa de competências (Pinto; Silva, 2019; Menegon *et al.*, 2016). Em áreas como Computação, em que a aprendizagem depende fortemente da experimentação, do erro como parte do processo e do desenvolvimento de autonomia intelectual, a centralidade do professor e a passividade do estudante tornam-se obstáculos significativos (Moresi *et al.*, 2019; Calderon; Silva; Feitosa, 2021). Por essa razão, a adoção de metodologias ativas vem sendo apontada como resposta às limitações desse modelo e como caminho para práticas mais investigativas, experimentais e centradas no aluno (Santos *et al.*, 2025).

Um dos pilares que sustentam as metodologias ativas é o protagonismo discente. Nessa abordagem, o estudante deixa de ser apenas receptor de informações e assume responsabilidade direta por seu processo de aprendizagem, tornando-se agente ativo na busca, interpretação e aplicação do conhecimento (Schwengber, 2024; Pinto; Silva, 2019). As atividades passam a ser estruturadas em torno de problemas, projetos, desafios e situações reais ou simuladas, o que estimula autonomia, tomada de decisão e reflexão crítica, além de favorecer a compreensão do

conteúdo em perspectiva aplicada (Moresi et al., 2019; Dias; Araújo, 2024). Consequentemente, o aprendizado torna-se mais significativo, pois o estudante constrói conhecimento por meio da prática e da interação, e não apenas pela memorização (Dias; Araújo, 2024).

Nesse cenário, o papel do professor também se ressignifica. O docente deixa de atuar exclusivamente como transmissor de conteúdo e passa a exercer função de mediador, orientador e facilitador do processo de aprendizagem. Sua atuação inclui propor desafios, estimular investigação, promover diálogo e acompanhar o percurso do estudante por meio de feedback contínuo, orientando a consolidação de conceitos e o amadurecimento das aprendizagens construídas (Pinto; Silva, 2019; Carmo et al., 2024; Genesio et al., 2024). Dessa forma, as metodologias ativas sustentam uma dinâmica pedagógica em que o estudante aprende pela experiência e pela reflexão, enquanto o professor organiza o ambiente de aprendizagem, conduz o processo e assegura coerência com os objetivos educacionais (Carmo et al., 2024).

No ensino de Computação, as contribuições das metodologias ativas se destacam por sua adequação às exigências próprias da área. O desenvolvimento de raciocínio lógico, pensamento algorítmico e capacidade de abstração; competências frequentemente relacionadas às dificuldades dos estudantes, tende a ser favorecido quando o processo de aprendizagem incorpora experiências práticas e significativas, alinhadas ao princípio do *learning by doing* (Pinto; Silva, 2019; Schwengber, 2024). Nesse sentido, estratégias como Aprendizagem Baseada em Problemas e Aprendizagem Baseada em Projetos são frequentemente associadas a maior engajamento, pois permitem que o estudante construa conhecimento durante a execução de atividades concretas, aproximando teoria e prática e atribuindo maior sentido ao conteúdo estudado (Dias; Araújo, 2024; Silva et al., 2018). Abordagens como Robótica Educacional e cultura maker também são mencionadas como formas de estimular criatividade, resiliência e envolvimento ativo do aluno, potencializando a formação por meio da experimentação (Schwengber, 2024).

Além disso, metodologias ativas favorecem o desenvolvimento de competências socioemocionais cada vez mais demandadas no contexto profissional contemporâneo, como colaboração, comunicação e trabalho em equipe. Estratégias como Aprendizagem Baseada em Equipes e gamificação incentivam interação, cooperação e engajamento, frequentemente com maior participação discente quando comparadas aos modelos tradicionais (Filho; Silva, 2024; Ferreira et al., 2016; Córdova-Esparza et al., 2024). O uso de elementos lúdicos, em particular, tem sido apontado como recurso pedagógico capaz de reduzir a carga cognitiva inicial em conteúdos abstratos e tornar a aprendizagem mais acessível, motivadora e contextualizada,

incluindo aplicações como computação desplugada, serious games e práticas gamificadas (Schwengber, 2024; Videnovik et al., 2023; Luccas, 2024).

A aprendizagem lúdica, por sua vez, pode ser compreendida como uma estratégia pedagógica que integra elementos de jogos ao processo educativo, conferindo dinamismo, motivação e significado à experiência de aprendizagem. Jogos educacionais conciliam objetivos pedagógicos com o caráter lúdico, favorecendo engajamento e estudo de conteúdos específicos (Oliveira et al., 2025). A ludicidade não se reduz à diversão: envolve dimensões cognitivas, sociais e emocionais e contribui para criatividade, sociabilidade e desenvolvimento integral do estudante (Menegon et al., 2016; Oliveira et al., 2025). Sob perspectiva histórica e cultural, Huizinga (2014) defende que o jogo antecede a própria cultura, ultrapassando a dimensão meramente física e assumindo valor simbólico e formativo, o que reforça sua relevância como ferramenta pedagógica.

No contexto da educação em Computação, em que os estudantes frequentemente estão habituados a ambientes interativos, o uso de jogos e práticas gamificadas pode aumentar níveis de envolvimento e participação, favorecendo atenção e concentração (Menegon et al., 2016; Genesio et al., 2024). A gamificação, entendida como o uso de elementos e mecânicas de jogos em contextos não lúdicos, busca motivar por meio de desafios, recompensas e feedbacks, podendo estimular desde socialização até superação de obstáculos e autorregulação da aprendizagem (Pinto; Silva, 2019; Filho; Silva, 2024; Menegon et al., 2016). Quando bem planejadas, essas estratégias podem conduzir o estudante a estados de elevada concentração e imersão, frequentemente associados ao conceito de flow, caracterizado por foco profundo, sensação de controle e distorção da percepção do tempo (Filho; Silva, 2024; Luccas, 2024).

Entre as abordagens lúdicas, os jogos de tabuleiro apresentam destaque por sua viabilidade prática no ambiente educacional e por sua aderência aos princípios das metodologias ativas. Esses jogos, classificados como não digitais, são reconhecidos como ferramentas pedagógicas de baixo custo, fácil aplicação e elevada adaptabilidade, especialmente úteis em contextos com restrições de infraestrutura tecnológica (Clementino et al., 2022; Silva et al., 2018). Além disso, sua natureza social favorece interação, cooperação, comunicação e resolução conjunta de problemas, habilidades relevantes tanto para a formação acadêmica quanto para a prática profissional em Computação (Menegon et al., 2016; Clementino et al., 2022). Experiências como o SoftGame, voltado ao ensino de Engenharia de Software, exemplificam como jogos analógicos podem transformar conteúdos teóricos em desafios estruturados, promovendo envolvimento ativo e aprendizagem significativa (Menegon et al., 2016; Clementino et al., 2022).

Em síntese, as metodologias ativas se mostram relevantes para o ensino de Computação ao favorecerem aprendizagem significativa, integração entre teoria e prática, protagonismo discente e desenvolvimento de competências técnicas e socioemocionais. Ao substituir a lógica de transmissão unilateral por estratégias investigativas, colaborativas e orientadas a desafios, essas metodologias possibilitam que o estudante deixe de ser um receptor passivo e atue como sujeito ativo na construção de conhecimento, sob mediação docente (Schwengber, 2024; Dias; Araújo, 2024; Carmo et al., 2024). Nesse conjunto, práticas lúdicas e jogos, especialmente jogos de tabuleiro, reforçam essa perspectiva ao ampliar motivação, engajamento e interação, oferecendo alternativas concretas para enfrentar dificuldades de aprendizagem e tornar o ensino de Computação mais participativo e acessível (Menegon et al., 2016; Clementino et al., 2022; Oliveira et al., 2025).

#### 1.4 Estado da arte

O estado da arte apresenta uma síntese de pesquisas recentes que investigam a aplicação de metodologias ativas e estratégias lúdicas no ensino de Computação, com foco na aprendizagem, no engajamento discente e na superação de dificuldades conceituais. A seguir, são discutidos trabalhos representativos da literatura, organizados de forma descritiva, evidenciando objetivos, abordagens e principais contribuições de cada estudo.

Almeida e Duarte (2020) apresentam o jogo de tabuleiro *Kaluanã* como um recurso pedagógico para auxiliar o ensino e a aprendizagem da disciplina de Algoritmos em cursos técnicos de Informática. O estudo parte do reconhecimento das dificuldades enfrentadas por estudantes iniciantes no desenvolvimento do raciocínio lógico e propõe uma abordagem lúdica baseada em estruturas fundamentais da programação, como variáveis, operadores, estruturas condicionais e de repetição. O jogo utiliza uma narrativa inspirada na mitologia indígena brasileira, promovendo interdisciplinaridade e engajamento, além de mecânicas competitivas que incentivam a tomada de decisão e a resolução de problemas. A aplicação em contexto real, com avaliação por questionários e observação direta, indica resultados positivos quanto à motivação, usabilidade e compreensão conceitual, reforçando o potencial dos jogos de tabuleiro como ferramentas didáticas eficazes no ensino de Algoritmos.

Gresse von Wangenheim et al. (2019) propõem o jogo de tabuleiro *SplashCode* como uma estratégia instrucional de baixo custo para o ensino de conceitos básicos de algoritmos e pensamento computacional na Educação Básica. O trabalho se destaca por adotar um processo sistemático de design instrucional, alinhado às diretrizes curriculares da SBC e da BNCC, além

de empregar o modelo MEEGA+KIDS para avaliação da experiência do jogador, usabilidade e aprendizagem. O jogo enfatiza a construção e execução de sequências de comandos, permitindo que os alunos “programem” movimentos de forma concreta e colaborativa. Os resultados do estudo de caso indicam elevada interação social, boa aceitação estética e evidências iniciais de aprendizagem, contribuindo para o fortalecimento de pesquisas que defendem o uso de jogos não digitais como alternativas viáveis e inclusivas para o ensino de Computação em contextos escolares com limitações de infraestrutura tecnológica.

Souza, Lima e Trindade (2025) investigam a aplicação do jogo de tabuleiro *Ábne døre e o Resgate da Magia* no ensino do Pensamento Computacional por meio da Computação Desplugada, com foco na habilidade de Decomposição prevista na BNCC (EF07CO05). O estudo utiliza uma sequência didática que combina Aprendizagem Baseada em Problemas (ABP) e Aprendizagem Baseada em Jogos (ABJ), aplicada a alunos do 7º ano do Ensino Fundamental II em uma escola pública. O jogo propõe desafios colaborativos nos quais os estudantes precisam decompor problemas em listas ou fluxogramas para avançar na narrativa e derrotar adversários. Os resultados apontam compreensão satisfatória do conceito, apesar de limitações relacionadas ao tempo e à mediação pedagógica, indicando que jogos de tabuleiro podem tornar o ensino do Pensamento Computacional mais acessível, lúdico e alinhado a práticas colaborativas de resolução de problemas

## **2 OBJETIVOS**

### **2.1 Objetivo Geral**

Desenvolver um jogo de tabuleiro educativo para apoiar o aprendizado das disciplinas de Engenharia da Computação na UFMA.

### **2.2 Objetivos Específicos**

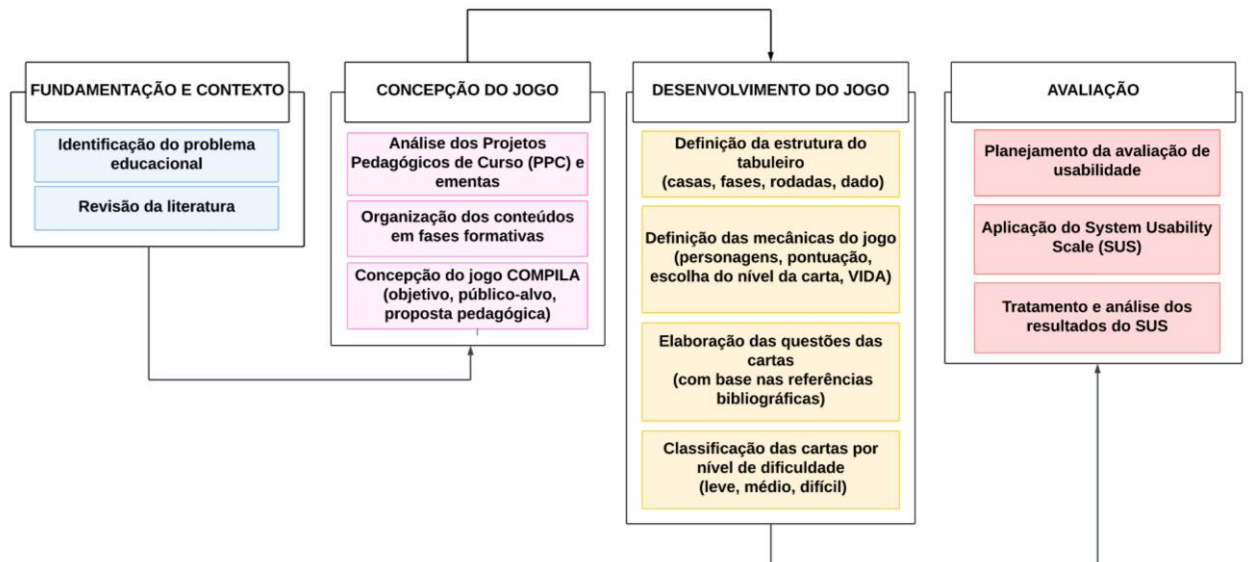
- a) Mapear os conteúdos previstos nas ementas das disciplinas do curso de Engenharia da Computação da UFMA a serem contemplados no jogo de tabuleiro, considerando diferentes níveis de complexidade;
- b) Projetar as mecânicas do jogo de tabuleiro, incorporando princípios de metodologias ativas e gamificação, de modo a favorecer o engajamento e a aprendizagem significativa dos estudantes;
- c) Desenvolver os elementos físicos e estruturais do jogo, incluindo tabuleiro, cartas e peças, assegurando coerência pedagógica com os conteúdos do curso e usabilidade adequada ao contexto acadêmico.

### 3 METODOLOGIA

O desenvolvimento metodológico do presente trabalho foi organizado em etapas sequenciais e inter-relacionadas, de modo a garantir coerência entre a fundamentação teórica, a concepção do jogo, o desenvolvimento de seus componentes e a avaliação da proposta. Essas etapas abrangem desde a identificação do problema educacional e a análise do contexto acadêmico, passando pela concepção e pelo desenvolvimento do jogo de tabuleiro COMPILA, até a realização da avaliação de usabilidade do recurso proposto.

O fluxo geral dessas etapas metodológicas encontra-se sintetizado no fluxograma apresentado na Figura 1, que ilustra de forma esquemática o processo adotado neste estudo.

**Figura 1** – Fluxograma do processo metodológico de desenvolvimento e avaliação do jogo COMPILA



Fonte: Compilação do autor, 2026.

#### 3.1 Tipo e abordagem da pesquisa

Este trabalho caracteriza-se como uma pesquisa aplicada, uma vez que se propõe ao desenvolvimento de um produto educacional voltado à aplicação prática no contexto do ensino superior em Engenharia da Computação. A pesquisa aplicada distingue-se por utilizar conhecimentos teóricos previamente consolidados com o objetivo de enfrentar problemas concretos e propor soluções em contextos institucionais específicos, mantendo rigor metodológico e relevância social.

Segundo Fleury e Werlang (2016), a pesquisa aplicada concentra-se na identificação de problemas presentes em organizações ou instituições e na busca de soluções que possam contribuir para a melhoria de práticas existentes. Nesse sentido, o presente estudo enquadra-se como pesquisa aplicada ao propor o desenvolvimento do jogo COMPILA como estratégia lúdica de apoio ao processo de ensino-aprendizagem, fundamentada em referenciais pedagógicos e curriculares.

Quanto à abordagem, a pesquisa adota uma perspectiva qualitativa, pois busca compreender aspectos relacionados à experiência do jogador, à usabilidade dos componentes físicos do jogo e à percepção dos participantes quanto à interação com o recurso proposto. Não se pretende realizar inferências estatísticas ou generalizações, mas sim analisar de forma descritiva e interpretativa os dados obtidos a partir da aplicação inicial do jogo.

No que se refere aos objetivos, o estudo apresenta caráter exploratório, uma vez que visa investigar o potencial do jogo COMPILA enquanto recurso pedagógico, permitindo identificar possibilidades, limitações e aspectos passíveis de aprimoramento em versões futuras. Esse caráter exploratório mostra-se adequado ao contexto de desenvolvimento de jogos educacionais, especialmente em etapas iniciais de concepção e avaliação.

### **3.2 Contexto da pesquisa e concepção do jogo COMPILA**

A presente pesquisa foi desenvolvida no contexto do ensino superior da Universidade Federal do Maranhão (UFMA), considerando os cursos de Bacharelado Interdisciplinar em Ciência e Tecnologia (BICT) e Engenharia da Computação. Esses cursos apresentam uma trajetória formativa articulada, na qual o BICT constitui, para parte dos estudantes, a etapa inicial da formação acadêmica, oferecendo fundamentos científicos e tecnológicos que subsidiam a progressão para cursos de engenharia.

Os Projetos Pedagógicos de Curso (PPC) e as ementas das disciplinas desses cursos foram adotados como documentos de referência para a compreensão da organização curricular e da progressão dos conteúdos ao longo da formação acadêmica. A análise desses documentos permitiu identificar os principais núcleos de conhecimento que estruturam o curso de Engenharia da Computação, bem como a evolução gradual da complexidade dos conteúdos abordados.

Com base na análise dos PPCs e das ementas das disciplinas, os conteúdos curriculares foram organizados em fases formativas, que não correspondem rigidamente aos períodos

acadêmicos formais. Essas fases representam agrupamentos didáticos de disciplinas definidos a partir de sua função pedagógica e do nível de complexidade exigido ao longo do curso.

Essa organização teve como objetivo abstrair a trajetória acadêmica do estudante em etapas progressivas, facilitando sua representação no jogo de tabuleiro educativo. Cada fase do jogo reúne disciplinas com características semelhantes, respeitando a lógica de progressão curricular e permitindo a simulação dos desafios típicos do percurso universitário.

A Figura 2 ilustra as fases do jogo COMPILA, evidenciando a progressão formativa desde os conteúdos introdutórios até a etapa final de síntese, correspondente à conclusão do curso. Essa representação visual foi elaborada com o propósito de sintetizar a análise curricular realizada, sem a pretensão de reproduzir integralmente os períodos acadêmicos oficiais.

**Figura 2** - Representação ilustrativa das fases do jogo COMPILA a partir da análise dos Projetos Pedagógicos de Curso do BICT e da Engenharia da Computação



Fonte: Compilação do autor, 2026.

O jogo COMPILA foi concebido como um jogo de tabuleiro educativo estruturado em rodadas sucessivas, no qual os participantes percorrem um tabuleiro composto por 50 casas, organizadas de forma sequencial. A primeira casa representa o ingresso no curso, enquanto a última corresponde à formatura, simbolizando, respectivamente, o início e a conclusão da trajetória acadêmica. O percurso do tabuleiro é dividido em oito fases, cada uma composta por seis casas, que representam diferentes momentos da formação ao longo da graduação. Essa organização busca refletir a progressão acadêmica vivenciada pelos estudantes, na qual novos conteúdos e desafios são apresentados de forma gradual e cumulativa.

A dinâmica da partida é limitada a um máximo de 20 rodadas, definidas como mecanismo de controle do tempo de jogo e como forma de simular as restrições temporais e os desafios progressivos enfrentados ao longo da graduação, como prazos, cargas de disciplinas e a necessidade de tomada de decisões em tempo limitado. Ao término desse limite, caso nenhum jogador alcance a etapa final do tabuleiro, a pontuação acumulada ao longo da partida é utilizada exclusivamente como critério de desempate, não havendo vencedores parciais durante o decorrer do jogo.

A progressão no tabuleiro está associada tanto à resolução de desafios relacionados aos conteúdos das disciplinas representadas em cada fase quanto ao deslocamento dos jogadores pelas casas do percurso. A movimentação é realizada por meio do lançamento de um dado de seis faces, utilizado inicialmente para a definição da ordem de início da partida, sendo o primeiro jogador aquele que obtiver o maior valor e, posteriormente, para determinar o número de casas a serem avançadas em cada turno, respeitando as regras estabelecidas e os efeitos decorrentes das cartas e demais componentes do jogo.

Cada fase do tabuleiro contempla 30 cartas dos três níveis de dificuldade: 10 de nível leve, 10 para nível médio e 10 para o nível difícil. Ao acessar uma casa correspondente a uma fase, o jogador tem a possibilidade de escolher o nível da carta que deseja responder, assumindo de forma consciente o grau de desafio associado à sua decisão. Essa mecânica introduz um elemento estratégico à dinâmica do jogo, permitindo ao participante ponderar entre desafios de menor complexidade conceitual e menor pontuação ou desafios mais complexos, associados a maior recompensa.

As cartas possuem valores de pontuação associados ao seu nível de dificuldade, sendo atribuídos 10 pontos para cartas de nível leve, 20 pontos para cartas de nível médio e 30 pontos para cartas de nível difícil. A pontuação acumulada ao longo da partida não define vencedores parciais, sendo utilizada exclusivamente como critério de desempate ao final das 20 rodadas, caso nenhum jogador alcance a casa final do tabuleiro.

Ressalta-se que o alcance da casa final (Formatura) não implica automaticamente a vitória do jogador. Para concluir a partida com êxito, o participante deve, obrigatoriamente, responder corretamente uma carta de nível difícil da última fase do jogo, representando a superação de um desafio final compatível com o encerramento da trajetória acadêmica. Caso as cartas de nível difícil da última fase tenham se esgotado, admite-se a utilização de cartas de nível difícil da fase imediatamente anterior, e assim sucessivamente, garantindo a continuidade da dinâmica do jogo sem comprometer o grau de desafio proposto. Essa regra reforça a ideia de que a conclusão do percurso acadêmico está condicionada não apenas ao avanço formal, mas também à superação de desafios conceituais finais.

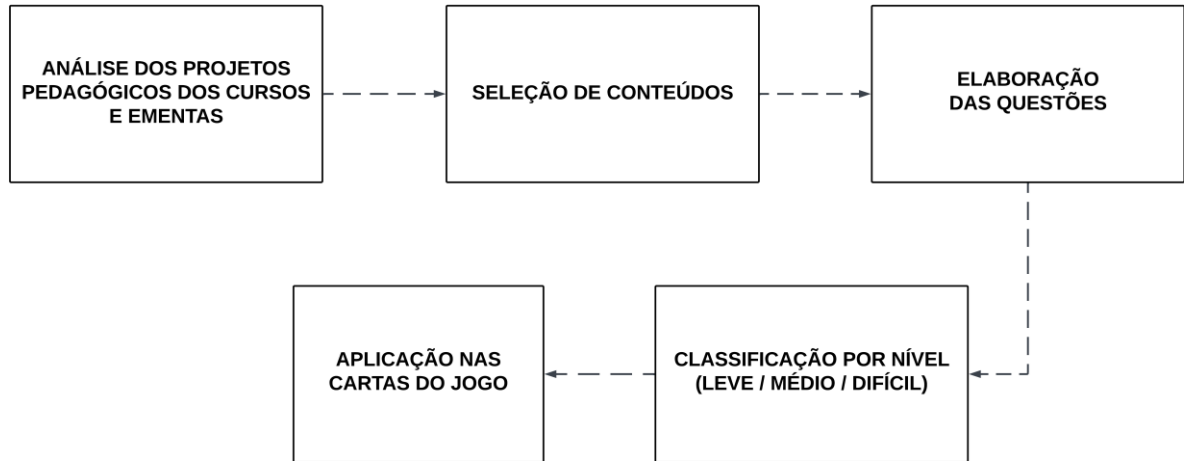
Os desafios são apresentados por meio de cartas de perguntas, cuja estrutura foi planejada para contemplar diferentes níveis de complexidade dos conteúdos abordados. O jogo utiliza componentes físicos, como tabuleiro, cartas, personagens e dado, que constituem a interface principal da experiência lúdica. Cada jogador é representado por um personagem, que atua como marcador de posição no tabuleiro e pode oferecer um efeito especial de uso único, acionado a critério do participante durante a partida. Os personagens são distribuídos de forma aleatória no início da partida, com o objetivo de preservar o equilíbrio competitivo do jogo e evitar vantagens estratégicas decorrentes da escolha livre. Os personagens disponíveis no jogo e seus respectivos efeitos especiais encontram-se descritos no Apêndice B. Essas habilidades foram concebidas como recursos estratégicos pontuais, permitindo lidar com situações adversas sem comprometer o equilíbrio geral da dinâmica do jogo.

Adicionalmente, pode-se incluir um componente opcional denominado VIDA, caracterizado como um personagem especial responsável por introduzir eventos aleatórios que impactam o andamento da partida, simulando imprevistos comumente enfrentados ao longo da trajetória acadêmica. Esse personagem possui um conjunto de 20 cartas, cujos efeitos podem tanto favorecer quanto sabotar os jogadores, influenciando direta ou indiretamente suas decisões e resultados. A utilização do componente VIDA é opcional e restrita a partidas com mais de dois participantes, de modo a preservar o equilíbrio e a dinâmica do jogo.

O processo de elaboração das cartas seguiu o fluxo metodológico apresentado na Figura 3, iniciando-se com a análise do Projeto Pedagógico do Curso (PPC) e das ementas das disciplinas contempladas no jogo. Essa etapa teve como objetivo identificar os conteúdos considerados centrais para cada fase do tabuleiro, respeitando a organização curricular do curso e a progressão natural dos conhecimentos ao longo da graduação. A partir dessa análise, procedeu-se à seleção dos conteúdos que serviriam de base para a formulação das questões, os quais foram fundamentados nos livros e materiais bibliográficos indicados nas ementas das

disciplinas correspondentes, assegurando alinhamento entre o jogo e as referências acadêmicas adotadas no curso.

**Figura 3** – Fluxo metodológico para elaboração das questões e cartas do jogo COMPILA.



**Fonte:** Compilação do autor, 2026.

Com base nesses conteúdos, foram elaboradas as questões que compõem as cartas do jogo, buscando equilibrar rigor conceitual e adequação ao contexto lúdico. As questões não foram concebidas com finalidade avaliativa formal, mas como instrumentos de estímulo ao raciocínio, à recuperação de conceitos e à articulação de conhecimentos ao longo da partida. Após a elaboração, as questões foram organizadas em três níveis de dificuldade (leve, médio e difícil), com o objetivo de estruturar os desafios de forma progressiva ao longo das fases do tabuleiro.

A definição dos níveis de dificuldade das cartas foi orientada pela complexidade conceitual dos conteúdos abordados e pelo grau de familiaridade esperado dos estudantes com esses temas ao longo da formação em Computação. Dessa forma, as cartas de nível leve concentram-se em conteúdos introdutórios ou mais recorrentes nas disciplinas, exigindo principalmente o reconhecimento de conceitos e definições fundamentais. As cartas de nível médio abordam conteúdos que demandam maior articulação conceitual, envolvendo relações entre tópicos, interpretação de situações ou aplicação de conhecimentos em contextos mais elaborados. Já as cartas de nível difícil contemplam conteúdos considerados mais densos ou abstratos, comumente associados a disciplinas avançadas ou a tópicos que historicamente apresentam maior grau de dificuldade para os estudantes.

Essa organização não tem como objetivo avaliar formalmente o desempenho acadêmico dos participantes, mas estruturar a dinâmica do jogo de modo progressivo, respeitando o percurso formativo representado no tabuleiro. A classificação em níveis busca, assim, oferecer desafios compatíveis com as diferentes fases da graduação, promovendo equilíbrio entre dificuldade, engajamento e experiência lúdica.

Ao permitir que o jogador escolha o nível da carta a ser respondida em cada fase, o jogo introduz um elemento de autonomia e tomada de decisão, incentivando o participante a refletir sobre sua disposição para enfrentar desafios mais complexos em troca de maiores recompensas. Tal mecânica reforça a proposta do COMPILA de representar, de forma simbólica, escolhas semelhantes às vivenciadas ao longo da trajetória acadêmica, como a seleção de disciplinas, o enfrentamento de conteúdos mais exigentes e a gestão dos próprios limites.

Dessa forma, a organização das cartas em níveis leve, médio e difícil constitui um recurso de design pedagógico e lúdico, alinhado à proposta do jogo e à representação da formação em Computação, contribuindo para uma progressão gradual dos desafios ao longo da partida, sem assumir caráter avaliativo formal. A relação completa das questões elaboradas, bem como seus respectivos referenciais bibliográficos, encontra-se apresentada no Apêndice A.

### **3.3 Procedimentos de Avaliação do Jogo COMPILA**

Após o desenvolvimento do jogo COMPILA, foi definida uma etapa de avaliação com o objetivo de analisar a usabilidade percebida e a experiência de uso do recurso pedagógico proposto. Considerando o contexto de evasão e retenção no ensino superior, especialmente em cursos da área de Computação, descrito na introdução deste trabalho, parte-se do pressuposto de que estratégias pedagógicas alternativas, como jogos educativos, somente podem contribuir para o engajamento discente se apresentarem regras claras, dinâmica compreensível e facilidade de interação por parte dos usuários.

Nesse sentido, a avaliação realizada não teve como finalidade mensurar impactos diretos sobre o desempenho acadêmico, a evasão ou a retenção discente, mas verificar se o jogo apresenta condições adequadas de usabilidade que justifiquem sua adoção como ferramenta pedagógica complementar em contextos educacionais. Tal abordagem é coerente com a proposta do trabalho, que se insere no campo das metodologias ativas e das abordagens lúdicas, nas quais a experiência do usuário desempenha papel central para o engajamento e a permanência acadêmica.

A partida do jogo COMPILA teve duração aproximada de 60 minutos, tempo suficiente para que os participantes percorressem diferentes fases do tabuleiro e interagissem com os principais elementos da dinâmica do jogo. O questionário SUS foi aplicado imediatamente após o término da partida, de modo a garantir que as respostas refletissem a experiência recente de uso do jogo. O formulário foi disponibilizado em formato impresso, sendo respondido individualmente pelos participantes.

Como instrumento de avaliação, foi adotado o System Usability Scale (SUS), um questionário padronizado amplamente utilizado em estudos de usabilidade por sua simplicidade, rapidez de aplicação e confiabilidade (Brooke, 2013). O SUS é composto por dez afirmações, respondidas em uma escala Likert de cinco pontos, que varia de “discordo totalmente” a “concordo totalmente”. As afirmações são organizadas de forma alternada entre itens positivos e negativos, com o objetivo de reduzir vieses de resposta e favorecer uma avaliação mais consistente da percepção dos usuários (Brooke, 2013).

A escolha do SUS justifica-se por sua ampla validação empírica e por sua aplicabilidade a diferentes tipos de sistemas e tecnologias, incluindo contextos não digitais, como jogos de tabuleiro. Além disso, estudos indicam que o instrumento apresenta resultados confiáveis mesmo quando aplicado a amostras reduzidas, característica que o torna adequado ao contexto deste trabalho (Brooke, 2013).

A avaliação foi realizada com a participação de seis estudantes, selecionados por conveniência, pertencentes à área de Computação. Dentre os participantes, quatro eram do sexo masculino e dois do sexo feminino, com idades entre 20 e 30 anos. Cinco estudantes eram regularmente matriculados no último período do curso de Engenharia da Computação, enquanto um participante era do sexo masculino, vinculado ao curso de Bacharelado Interdisciplinar em Ciência e Tecnologia (BICT), encontrando-se no quinto período. Esse perfil está alinhado ao público-alvo do jogo COMPILA.

Após a participação em uma partida do jogo, os estudantes responderam individualmente ao questionário System Usability Scale (SUS), de forma voluntária e anônima. Não houve coleta de dados pessoais ou qualquer forma de identificação dos respondentes, garantindo que as respostas refletissem exclusivamente a experiência prática com o jogo. O formulário completo do questionário encontra-se apresentado no Apêndice C.

As respostas obtidas foram tratadas conforme o método de cálculo proposto originalmente por Brooke (1996), no qual as pontuações dos itens são ajustadas de acordo com sua polaridade e somadas, resultando em um valor final multiplicado por um fator de correção.

O resultado obtido varia de 0 a 100 e representa uma medida global da usabilidade percebida, não devendo ser interpretado como percentual (Brooke, 2013).

## 4 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados obtidos a partir do desenvolvimento e da avaliação do jogo de tabuleiro educativo COMPILA. Inicialmente, é apresentado o produto desenvolvido, seguido pelos resultados da avaliação de usabilidade realizada por meio do System Usability Scale (SUS). Por fim, os resultados são discutidos à luz dos objetivos propostos neste trabalho, considerando o caráter exploratório da pesquisa.

O principal produto desenvolvido neste trabalho foi o jogo de tabuleiro educativo COMPILA, cuja materialização visual é ilustrada na Figura 4. A figura ilustra o tabuleiro resultante do processo de concepção e desenvolvimento descrito no capítulo metodológico, permitindo a visualização do produto e de sua organização gráfica enquanto recurso pedagógico proposto neste estudo.

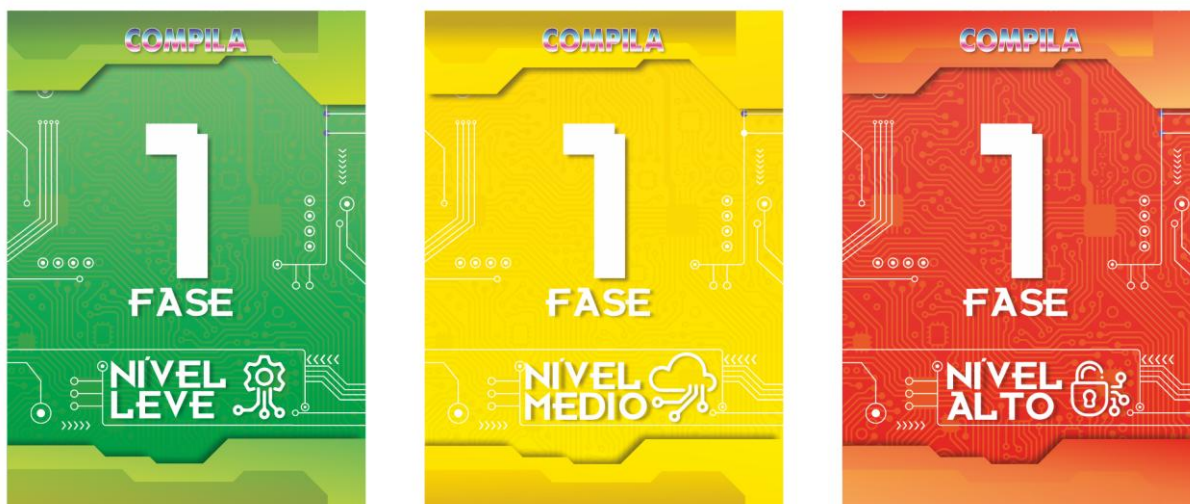
**Figura 4** – Tabuleiro do jogo COMPILA



Fonte: Compilação do autor, 2026.

A Figura 5 ilustra a frente das cartas do jogo COMPILA, evidenciando a identificação visual da fase e do nível de dificuldade correspondente. Observa-se o uso de elementos gráficos e cromáticos distintos para diferenciar os níveis, contribuindo para a organização visual do jogo e para o reconhecimento imediato das cartas durante a partida.

**Figura 5** – Frente das cartas do jogo COMPILA



**Fonte:** Compilação do autor, 2026.

A Figura 6 ilustra o verso das cartas do jogo COMPILA, no qual são exibidos os desafios propostos aos jogadores. Observa-se a organização visual do conteúdo das questões, bem como a indicação dos efeitos associados às respostas. Destaca-se ainda que a alternativa correta se encontra posicionada discretamente em um dos cantos da carta, apresentada de forma invertida, permitindo sua consulta apenas quando necessário, sem interferir na dinâmica principal da partida. As cartas ilustradas correspondem ao resultado do processo de elaboração descrito no capítulo metodológico, integrando os elementos pedagógicos do jogo.

**Figura 6** – Verso das cartas do jogo COMPILA



**Fonte:** Compilação do autor, 2026.

A Figura 7 elucida os personagens desenvolvidos para o jogo COMPILA, os quais compõem os elementos visuais e estratégicos do produto. Cada personagem é representado graficamente de forma distinta, contribuindo para a identidade visual do jogo e para a diferenciação entre os participantes durante a partida. Os personagens ilustrados correspondem ao resultado do processo de concepção descrito no capítulo metodológico, integrando a proposta lúdica do jogo desenvolvido neste estudo.

**Figura 7** – Personagens do jogo COMPILA

Fonte: Compilação do autor, 2026.

A partir da apresentação dos componentes que constituem o jogo COMPILA (incluindo tabuleiro, cartas e personagens), conclui-se a descrição do recurso pedagógico desenvolvido neste trabalho enquanto produto educacional. Esses elementos materializam a proposta metodológica adotada, resultante de um processo sistemático de concepção, desenvolvimento e organização, conforme descrito no capítulo metodológico.

Considerando que a efetividade de recursos lúdicos no contexto educacional está diretamente relacionada à clareza de suas regras, à compreensão de sua dinâmica e à facilidade de interação por parte dos usuários, torna-se necessário avaliar a usabilidade do jogo desenvolvido. Nesse sentido, a etapa seguinte deste capítulo dedica-se à apresentação e à análise dos resultados obtidos por meio da aplicação do *System Usability Scale (SUS)*, instrumento amplamente utilizado para a avaliação da usabilidade percebida de sistemas e produtos, incluindo jogos e recursos educacionais.

As respostas obtidas foram tratadas de acordo com o método de cálculo proposto por Brooke (1996), no qual os itens do questionário recebem ajustes específicos conforme sua polaridade, sendo posteriormente somados e multiplicados por um fator de correção. A partir desse procedimento, foi obtida uma pontuação média de 85,8 pontos no SUS. O detalhamento

das respostas individuais dos participantes ao questionário encontra-se apresentado no Apêndice D.

A Tabela 1 apresenta o resultado consolidado da avaliação de usabilidade do jogo COMPILA, considerando o número de participantes envolvidos e a pontuação média obtida no instrumento.

Tabela 1 – Resultado da avaliação de usabilidade do jogo COMPILA (SUS)

<b>Avaliação</b>	<b>Nº de participantes</b>	<b>Pontuação média SUS</b>
<b>COMPILA</b>	06	85,8

Fonte: Compilação do autor, 2026.

De acordo com os critérios de interpretação do SUS, pontuações superiores a 68 indicam níveis de usabilidade acima da média, enquanto valores superiores a 80 são associados a sistemas com boa aceitação e elevada facilidade de uso (Brooke, 2013). Nesse sentido, o resultado obtido sugere que o jogo COMPILA apresenta alto nível de usabilidade percebida, indicando que os participantes consideraram o jogo de fácil compreensão e utilização.

Ressalta-se que os resultados apresentados se referem exclusivamente à usabilidade do jogo enquanto recurso lúdico, não tendo como objetivo mensurar impactos diretos sobre aprendizagem, desempenho acadêmico ou permanência discente. Ainda assim, os dados obtidos indicam que o jogo possui condições adequadas de uso que justificam sua adoção como ferramenta pedagógica complementar em contextos educacionais, especialmente no âmbito de propostas fundamentadas em metodologias ativas e abordagens lúdicas.

Como limitações deste estudo, destaca-se o número reduzido de participantes envolvidos na avaliação de usabilidade, bem como a utilização de um único instrumento de coleta de dados. Tais limitações não invalidam os resultados obtidos, mas indicam possibilidades para investigações futuras com amostras ampliadas e abordagens complementares.

## 5 CONCLUSÃO

Este trabalho teve como objetivo o desenvolvimento de um jogo de tabuleiro educativo, denominado COMPILA, voltado ao apoio do processo de ensino-aprendizagem em cursos da área de Computação, com foco específico na Engenharia da Computação da Universidade Federal do Maranhão (UFMA). A proposta surgiu a partir da constatação, amplamente discutida na literatura, de que cursos das áreas de Ciências Exatas, Engenharias e Computação apresentam elevados índices de reprovação, retenção e evasão, especialmente nos períodos iniciais da graduação, associados a dificuldades conceituais, alto grau de abstração e práticas pedagógicas predominantemente tradicionais.

Ao longo do trabalho, foi possível contextualizar esses fenômenos, evidenciando que a evasão e a retenção não podem ser compreendidas apenas como responsabilidade individual do estudante, mas como resultados de fatores multifatoriais, que envolvem aspectos institucionais, pedagógicos, curriculares e socioeconômicos. Nesse cenário, as metodologias ativas de aprendizagem e as abordagens lúdicas foram discutidas como alternativas pedagógicas relevantes, capazes de promover maior engajamento discente, aprendizagem significativa e protagonismo do estudante, especialmente em áreas que demandam raciocínio lógico, pensamento algorítmico e capacidade de abstração.

Com base no referencial teórico, o jogo COMPILA foi concebido como um recurso pedagógico complementar, estruturado para representar, de forma simbólica e lúdica, a trajetória acadêmica em cursos de Computação. A organização do tabuleiro em fases formativas, a elaboração das cartas a partir das ementas e referências bibliográficas do curso, bem como a definição de níveis progressivos de dificuldade, permitiram alinhar o jogo à lógica curricular da Engenharia da Computação. Além disso, a inclusão de elementos como personagens com habilidades especiais e eventos aleatórios buscou enriquecer a dinâmica do jogo, promovendo tomada de decisão, interação social e reflexão sobre desafios recorrentes da vida acadêmica.

No que se refere à avaliação do recurso desenvolvido, optou-se pela aplicação do System Usability Scale (SUS), instrumento amplamente validado na literatura para análise de usabilidade percebida. Os resultados obtidos indicaram uma pontuação média de 85,8, valor considerado elevado segundo os critérios de interpretação do SUS, sugerindo que o jogo COMPILA apresenta boa aceitação, regras compreensíveis e facilidade de uso. Embora a avaliação não tenha tido como objetivo mensurar impactos diretos sobre aprendizagem ou redução de evasão e retenção, os dados apontam que o jogo possui condições adequadas de

usabilidade que justificam sua adoção como ferramenta pedagógica complementar em contextos educacionais.

É importante destacar que este estudo apresenta limitações, especialmente relacionadas ao número reduzido de participantes na etapa de avaliação e à utilização de um único instrumento de coleta de dados. Tais limitações não invalidam os resultados alcançados, mas indicam a necessidade de investigações futuras que envolvam amostras ampliadas, aplicações em diferentes contextos acadêmicos e o uso de instrumentos complementares, como análises qualitativas, observação em sala de aula e estudos longitudinais sobre engajamento e aprendizagem.

Como trabalhos futuros, sugere-se a aplicação do jogo COMPILA em disciplinas específicas da graduação, a fim de investigar de forma mais aprofundada seus impactos no processo de aprendizagem, no engajamento discente e na percepção dos estudantes sobre os conteúdos abordados. Além disso, adaptações do jogo para outros cursos da área de Computação ou para diferentes níveis de ensino podem ampliar seu potencial de utilização e contribuição acadêmica.

Por fim, conclui-se que o COMPILA se configura como um produto educacional coerente com os princípios das metodologias ativas e da aprendizagem lúdica, oferecendo uma alternativa viável, de baixo custo e alinhada ao contexto institucional para apoiar o ensino de Computação. Ao integrar fundamentos teóricos, organização curricular e elementos lúdicos, o jogo contribui para reflexões sobre práticas pedagógicas inovadoras e reforça a importância de estratégias que valorizem o estudante como sujeito ativo no processo de construção do conhecimento.

## REFERÊNCIAS

ADACHI, Ana Amelia Chaves Teixeira. **Evasão e evadidos nos cursos de graduação da UFMG**. 2009. Dissertação (Mestrado em Educação) – Universidade Federal de Minas Gerais, Belo Horizonte, 2009. Disponível em: <https://repositorio.ufmg.br/items/fec2b7f0-96ca-46b0-9068-f41b92f383cf>. Acesso em: 15 nov. 2025.

ALMEIDA, Thiago Fernandes de; DUARTE, Eduardo Seperuelo. Kaluanã: um jogo de tabuleiro como recurso para o auxílio do ensino e aprendizagem de algoritmos. **Revista de Educação, Ciências e Matemática**, Rio de Janeiro, v. 10, n. 2, p. 246–258, 2020. ISSN 2238-2380.

ARAÚJO, Ana Cléssia; MARIANO, Francisca; DE OLIVEIRA, Celina. **Determinantes acadêmicos da retenção no Ensino Superior**. Ensaio: Avaliação e Políticas Públicas em Educação, Rio de Janeiro, v. 29, ed. 113, p. 1045–1066, out./dez. 2021. Disponível em: <https://www.scielo.br/j/ensaio/a/zf9fLzPYq4tJ543zs6crQmN/?lang=pt>. Acesso em: 15 nov. 2025.

ARIMOTO, Maurício; OLIVEIRA, Weldrey. Dificuldades no Processo de Aprendizagem de Programação de Computadores: um Survey com Estudantes de Cursos da Área de Computação. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 27., 2019, Belém. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2019. p. 244-254. ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2019.6633>.

BARCELOS, Thiago Schumacher; FRANGO, Ismar. Pensamento Computacional e Educação Matemática: relações para o ensino de computação na educação básica. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 2012, **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2012. Disponível em: <https://sol.sbc.org.br/index.php/wei/article/view/29076>. Acesso em: 15 nov. 2025.

BARROSO, M. F.; FALCÃO, E. B. M. Evasão universitária: o caso do Instituto de Física da UFRJ. In: ENCONTRO NACIONAL DE PESQUISA EM ENSINO DE FÍSICA (ENPEF), 9., 2004, Jaboticatubas, MG, Brasil. **Anais [...]**. Jaboticatubas: Sociedade Brasileira de Física, 2004. Disponível em: <https://www.if.ufrj.br/~carlos/pef/materiais/marta-epf2004-evasao-co12-2.pdf>. Acesso em: 15 nov. 2025.

BRASIL. Ministério da Educação. **Portaria nº 1.117, de 14 de agosto de 2013**. Institui o Programa Nacional de Assistência Estudantil (PNAES). Diário Oficial da União: seção 1, Brasília, DF, 15 ago. 2013. Disponível em: [https://portal.mec.gov.br/arquivos/pdf/portaria\\_pnaes.pdf](https://portal.mec.gov.br/arquivos/pdf/portaria_pnaes.pdf). Acesso em: 15 nov. 2025.

BRITO, Palloma Stéphanne Silva. **O uso de ferramentas computacionais para o ensino de programação para alunos de engenharia**. 2019. 38 f. Monografia (Graduação em Ciência

da Computação) – Universidade Federal de Ouro Preto, Ouro Preto, 2019. Disponível em: <https://www.monografias.ufop.br/handle/35400000/2274>. Acesso em: 15 nov. 2025.

BOSSE, Yorah. **Padrões de dificuldades relacionadas com o aprendizado de programação.** 2020. Tese (Doutorado em Ciência da Computação) – Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2020. DOI: <https://doi.org/10.11606/T.45.2020.tde-14072020-172808>. Disponível em: <https://www.teses.usp.br/teses/disponiveis/45/45134/tde-14072020-172808/>. Acesso em: 15 nov. 2025.

BOUCINHA, Rafael Marimon. **Aprendizagem do pensamento computacional e desenvolvimento do raciocínio.** 2017. 151 f. Tese (Doutorado em Informática na Educação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017. Disponível em: <http://hdl.handle.net/10183/172300>. Acesso em: 15 nov. 2025.

BROOKE, John. **SUS: a “quick and dirty” usability scale.** In: JORDAN, P. W.; THOMAS, B.; WEERDMEESTER, B. A.; MCCLELLAND, A. L. (org.). *Usability evaluation in industry.* London: Taylor & Francis, 1996. p. 189–194.

BROOKE, John. **SUS: a retrospective.** *Journal of Usability Studies*, v. 8, n. 2, p. 29–40, fev. 2013.

CALDERON, Ivanilse; SILVA, Williamson; FEITOSA, Eduardo. Um mapeamento sistemático da literatura sobre o uso de metodologias ativas durante o ensino de programação no Brasil. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 32., 2021, Porto Alegre. **Anais [...].** Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 1152-1161. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/18138>. Acesso em: 15 nov. 2025.

CAMPELLO, Antonio de Vasconcellos Carneiro; LINS, Luciano Nadler. Metodologia de análise e tratamento da evasão e retenção em cursos de graduação de instituições federais de ensino superior. In: XXVIII ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO (ENEGEP), Rio de Janeiro, RJ, Brasil, 13 a 16 out. 2008. **Anais [...].** Rio de Janeiro: ABEPRO, 2008. p 1-13. Disponível em: [http://abepro.org.br/biblioteca/enegep2008\\_TN\\_STO\\_078\\_545\\_11614.pdf](http://abepro.org.br/biblioteca/enegep2008_TN_STO_078_545_11614.pdf). Acesso em: 16 nov. 2025.

CARMO, Edenilson Carlos Barros do; SEGUNDO, Fagno Sousa Oliveira; FRAZÃO, Mariane Rafaela Coelho et al. Metodologias ativas no ensino lúdico de políticas de segurança de redes e firewall. **Revista Sociedade Científica**, v. 7, n. 1, p. 2494-2506, 03 jun. 2024. DOI: <https://doi.org/10.61411/rsc202451117>. Disponível em:

<https://revista.scientificsociety.net/wp-content/uploads/2024/06/Art.140-2024.pdf>. Acesso em: 16 nov. 2025.

CARRETTA, Marcelo La. Como fazer jogos de tabuleiro: manual prático. In: SIMPÓSIO BRASILEIRO DE GAMES E ENTRETENIMENTO DIGITAL (SBGames), 17., 2018, Foz do Iguaçu. **Proceedings...** Porto Alegre: Sociedade Brasileira de Computação, 2018. p. 1621–1627. Tutorial.

CLEMENTINO, Eduardo G et al. Jogos não digitais para ensino de computação – um mapeamento sistemático. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 33., 2022. **Anais...**, p. 540–550, 2022. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/22438>. Acesso em: 16 nov. 2025.

CÓRDOVA-ESPARZA, Diana-Margarita; ROMERO-GONZÁLEZ, Julio-Alejandro; CÓRDOVA-ESPARZA, Karen-Edith; TERVEN, Juan; LÓPEZ-MARTÍNEZ, Rocío-Edith. **Active learning strategies in computer science education: a systematic review**. Multimodal Technologies and Interaction, Basel, v. 8, n. 6, p. 50, 13 jun. 2024. DOI: <https://doi.org/10.3390/mti8060050>. Disponível em: <https://www.mdpi.com/2414-4088/8/6/50>. Acesso em: 16 nov. 2025.

CUNHA, J. V. A.; NASCIMENTO, E. M.; DURSO, S. O. **Razões e influências para a evasão universitária: um estudo com estudantes ingressantes nos cursos de Ciências Contábeis de instituições públicas federais da região Sudeste**. Advances in Scientific and Applied Accounting, São Paulo, v. 9, n. 2, p. 141–161, maio/ago. 2016. Disponível em: <https://asaa.anpcont.org.br/index.php/asaa/article/view/260>. Acesso em: 16 nov. 2025.

DIAS, Helder Daniel de Azevedo; ARAUJO, Josivaldo de Souza. Um relato de experiência do uso de metodologias ativas na construção do pensamento computacional paralelo. **Revista Brasileira de Informática na Educação**, Porto Alegre, v. 32, p. 1–10, 2024.

FERREIRA, Luciana; INOCÊNCIO, Ana Carolina Gondim; AFONSO, Paulo; LOPES, Márcio Moraes. Gamificação aplicada ao ensino de gerência de projetos de software. In: WORKSHOP DE INFORMÁTICA NA ESCOLA (WIE), 25., 2016, Belém. **Anais...** Porto Alegre: Sociedade Brasileira de Computação, 2016. p. XX–YY. DOI: 10.5753/cbie.wie.2016.151. Disponível em: <https://sol.sbc.org.br/index.php/wie/article/view/16411>. Acesso em: 16 nov. 2025.

FILHO, Jesse Nery; SILVA, Fabianne Costa da. Gamificação para além das salas de aula: metodologias ativas para professores contemporâneos. **RE@D – Revista de Educação a Distância e eLearning**, v. 7, n. 2, e202433, dez. 2024. DOI: <https://doi.org/10.34627/redvol7iss2e202433>. Disponível em: [https://revistas.rcaap.pt/lead\\_read/article/view/37906](https://revistas.rcaap.pt/lead_read/article/view/37906). Acesso em: 16 nov. 2025.

FLEURY, Maria Tereza Leme; WERLANG, Sérgio R. C. **Pesquisa aplicada: conceitos e abordagens.** *APGv Pesquisa*, p. 10-15, 2017. Disponível em: <https://periodicos.fgv.br/apgvpesquisa/article/view/72796/69984>. Acesso em: 17 nov. 2025.

FRANZEN, Evandro. **Estratégia de ensino e aprendizagem ativa aplicada ao aprendizado de algoritmos e programação: identificação e análise da motivação dos estudantes.** 2019. 232 f. Tese (Doutorado em Informática na Educação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2019. Disponível em: <http://hdl.handle.net/10183/202383>. Acesso em: 17 nov. 2025.

GAIOSO, Natália Pacheco de Lacerda. **A evasão discente na educação superior no Brasil: na perspectiva de alunos e dirigentes.** 95p. Dissertação (Mestrado em Educação). Universidade Católica de Brasília. BrasíliaDF, 2005.

GENESIO, Nikolas Oliver Sales; OLIVEIRA, Alessandra Marta de; OLIVEIRA, Edmar Wellington; VALLE, Pedro Henrique Dias. Panorama de estudos sobre jogos educacionais digitais em educação em computação. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 32., 2024, Porto Alegre. **Anais do Workshop sobre Educação em Computação.** Porto Alegre: Sociedade Brasileira de Computação, 2024. p. 1–13. Disponível em: <https://sol.sbc.org.br/index.php/wei/article/view/29672>. Acesso em: 17 nov. 2025.

GRESSE VON WANGENHEIM, Christiane; MEDEIROS, Giselle Araújo e Silva de; MISSFELDT FILHO, Raul; PETRI, Giani; PINHEIRO, Fernando da Cruz; FERREIRA, M. Nathalie F.; HAUCK, Jean C. R. Desenvolvimento e avaliação de um jogo de tabuleiro para ensinar o conceito de algoritmos na educação básica. **Revista Brasileira de Informática na Educação (RBIE)**, Porto Alegre, v. 27, n. 3, p. 310–335, 2019. DOI: 10.5753/RBIE.2019.27.03.310.

HOED, Raphael Magalhães. **Análise da evasão em cursos superiores: o caso da evasão em cursos superiores da área de Computação.** 2016. 188 f. Dissertação (Mestrado Profissional em Computação Aplicada) – Universidade de Brasília, Brasília, 2016. Disponível em: [https://repositorio.unb.br/bitstream/10482/22575/1/2016\\_RaphaelMagalh%C3%A3esHoed.pdf](https://repositorio.unb.br/bitstream/10482/22575/1/2016_RaphaelMagalh%C3%A3esHoed.pdf). Acesso em: 17 nov. 2025.

HUIZINGA, Johan. **Homo ludens: o jogo como elemento da cultura.** 8. ed. São Paulo: Perspectiva, 2014.

JUNIOR, Aldo Candeia de Albuquerque et al. Um estudo de caso sobre evasão no curso presencial de computação da Universidade Estadual da Paraíba – Campus VII – Patos – PB. In: CONEDU – Congresso Nacional de Educação, IX, 2023, Campina Grande, PB, Brasil.

Anais [...]. Campina Grande: Realize Editora, 2023. Disponível em: <https://editorarealize.com.br/artigo/visualizar/101391>. Acesso em: 17 nov. 2025.

LIMA, Henrique Targino de. **Análise dos fatores que influenciam a taxa de sucesso na graduação em universidades federais**. 2024. Trabalho de Conclusão de Curso (Bacharelado em Ciências Contábeis) – Universidade Federal do Rio Grande do Norte, Natal, 2024. Disponível em: <https://repositorio.ufrn.br/items/9e750d23-2c54-4ba8-a1a0-9457891b7585>. Acesso em: 17 nov. 2025.

LUCCAS, Matheus dos Santos. **Desenvolvimento de uma metodologia para elaboração de jogos educacionais para ensino de Computação**. 2024. 164 f. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2024. Disponível em: [https://www.teses.usp.br/teses/disponiveis/55/55134/tde-16042024-091707/publico/MatheusdosSantosLuccas\\_ME\\_revisada.pdf](https://www.teses.usp.br/teses/disponiveis/55/55134/tde-16042024-091707/publico/MatheusdosSantosLuccas_ME_revisada.pdf). Acesso em: 17 nov. 2025.

MEC. Diplomação, retenção e evasão nos cursos de graduação em instituições de ensino superior públicas: resumo do relatório apresentado a ANDIFES, ABRUEM e SESu/MEC pela Comissão Especial. **Avaliação: Revista da Avaliação da Educação Superior**, Campinas; Sorocaba, v. 1, n. 2, p. 55–65, 1996. Disponível em: <https://periodicos.uniso.br/avaliacao/article/view/739>. Acesso em: 18 nov. 2025.

MELO, Geovana F.; NAVES, Marisa L. de P. Retenção e evasão: desafios para a gestão da educação superior. *Revista FORGES – Fórum da Gestão do Ensino Superior nos Países e Regiões de Língua Portuguesa*, v. 5, n. 60, p. [indicar intervalo de páginas se disponível], 2017. Disponível em: <https://publicacoes.riqual.org/wp-content/uploads/2022/04/Forges-5-60.pdf>. Acesso em: 18 nov. 2025.

MENEGON, Stela Ronchi. **Softgame**: jogo de tabuleiro como material de apoio à engenharia de software. 2016. Trabalho de Conclusão de Curso (Graduação em Tecnologias da Informação e Comunicação) – Universidade Federal de Santa Catarina, Campus Araranguá, 2016. Disponível em: <https://repositorio.ufsc.br/handle/123456789/166322>. Acesso em: 18 nov. 2025.

MOONEY, O.; PATTERSON, V.; O’CONNOR, M.; CHANTLER, A. **A Study of Progression in Irish Higher Education**. Dublin: Higher Education Authority, 2010. Disponível em: <https://hea.ie/assets/uploads/2017/06/A-Study-of-Progression-in-Higher-Education.pdf>. Acesso em: 18 nov. 2025.

MORESI, Eduardo Amadeu Dutra; BRAGA FILHO, Mário de Oliveira; HARTMANN, Wilson Carlos; CARVALHO, Christine Maria Soares de. Gamificação como metodologia ativa: estudo de caso na disciplina de Engenharia de Software. **Revista Iberoamericana de**

**Sistemas, Cibernética e Informática – RISCI**, v. 16, p. 63-68, 2019. Disponível em: <https://www.iisci.org/journal/pdv/risci/pdfs/CA439UN19.pdf>. Acesso em: 09 nov. 2025.

NERY FILHO, José. Gamificação na Educação: Ensino de metodologias ativas no Curso de Licenciatura em Ciências da Computação. **RE@D – Revista de Educação a Distância e eLearning**, v. 7, n. 2, 2024. Disponível em: [https://revistas.rcaap.pt/lead\\_read/article/view/36596/25648](https://revistas.rcaap.pt/lead_read/article/view/36596/25648). Acesso em: 17 nov. 2025.

NETO, Carlos Catini. **Um método baseado em aprendizagem corporificada para o ensino de conceitos de programação orientada a objetos**. 2025. 122 f. Dissertação (Mestrado Profissional em Ciência da Computação) – Centro Universitário Campo Limpo Paulista – UNIFACCAMP, Campo Limpo Paulista, 02 abr. 2025. Disponível em: <http://hdl.handle.net/10183/202383>. Acesso em: 09 nov. 2025.

NETO, João José. A teoria da computação e o profissional de informática. **Revista de Computação e Tecnologia (ReCeT)**, v. 1, n. 1, 28 out. 2009. Disponível em: <https://revistas.pucsp.br/index.php/ReCET/article/view/1572>. Acesso em: 10 nov. 2025.

OLIVEIRA, Gabriel; ROCHA, Pedro; DUARTE, Paulina; LIMA, Joanne; CASTRO, Carlos; PAIVA, Anselmo Cardoso de; MAIA, Ivana; RIVERO, Luis. Jogos educacionais para o ensino de computação aplicados no Brasil: estado atual e perspectivas futuras. **Novas Tecnologias na Educação (RENOTE)**, v. 23, n. 1, p. 463–475, 2025. DOI: <https://doi.org/10.22456/1679-1916.149290>. Disponível em: <https://seer.ufrgs.br/index.php/renote/article/view/149290>. Acesso em: 18 nov. 2025.

PAULA, Maria de Fátima Costa de. Políticas de democratização da educação superior brasileira: limites e desafios para a próxima década. **Avaliação: Revista da Avaliação da Educação Superior**, Campinas, v. 22, n. 2, p. 301–315, jul./dez. 2017. Disponível em: <https://www.scielo.br/j/aval/a/KYs6H9L5YpppTCZHpHGd8SK/?lang=pt>. Acesso em: 18 nov. 2025.

PINTO, Fernando S.; SILVA, Paulo C. EduGamification: uma metodologia de gamificação para apoiar o processo ensino-aprendizagem. **Anais do Workshop sobre Educação em Computação (WEI)**, 2019, p. 414-428. Disponível em: <https://sol.sbc.org.br/index.php/wei/article/view/6647>. Acesso em: 18 nov. 2025.

PREDES, Fabiane da Silva de Lemos; PREDES JUNIOR, Ademir Ribeiro. A implementação do Programa de Apoio a Planos de Reestruturação e Expansão das Universidades Federais (REUNI): um estudo de caso da Universidade Federal Rural do Rio de Janeiro. In: COLÓQUIO INTERNACIONAL DE GESTÃO UNIVERSITÁRIA (CIGU), XIV, 2014, Florianópolis. **Anais [...]**. Florianópolis: Universidade Federal de Santa Catarina – UFSC,

2014. Disponível em: <https://repositorio.ufsc.br/bitstream/handle/123456789/131921/2014-264.pdf?sequence=1&isAllowed=y>. Acesso em: 18 nov. 2025.

RAPKIEWICZ, Clevi Elena; FALKEMBACH, Gilse; SEIXAS, Louise; ROSA, Núbia dos Santos; CUNHA, Vanildes Vieira da; KLEMANN, Miriam. **Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais**. Novas Tecnologias na Educação, Porto Alegre, v. 4, n. 2, p. 1–11, dez. 2006. Disponível em: <https://lume.ufrgs.br/bitstream/handle/10183/22862/000625846.pdf>. Acesso em: 18 nov. 2025.

SACCARO, Alice; FRANÇA, Marco Túlio Aniceto; JACINTO, Paulo de Andrade. Fatores associados à evasão no ensino superior brasileiro: um estudo de análise de sobrevivência para os cursos das áreas de Ciência, Matemática e Computação e de Engenharia, Produção e Construção em instituições públicas e privadas. **Estudos Econômicos**, São Paulo, v. 49, n. 2, p. 337–373, 2019. Disponível em: <https://www.scielo.br/j/ee/a/9YxHxWkk6Dzy35CpgmxXbPt/?lang=pt>. Acesso em: 16 nov. 2025.

SANTOS, Marcelo Silva. **Uma análise probabilística da retenção na Universidade Federal da Bahia: um estudo de caso no curso de Ciência da Computação**. 2016. 109 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal da Bahia, Salvador, 2016. Disponível em: <https://repositorio.ufba.br/handle/ri/19379>. Acesso em: 13 nov. 2025.

SANTOS, Rodolfo Sanches Saraiva dos. **Evasão escolar universitária e estratégias de intervenções para retenção do estudante: um estudo de caso na Universidade Federal de São Carlos**. 2022. 191 f. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional) – Universidade de São Paulo, São Carlos, 2022. DOI: <https://doi.org/10.11606/D.55.2022.tde-21092022-094249>. Disponível em: <https://www.teses.usp.br/teses/disponiveis/55/55134/tde-21092022-094249/>. Acesso em: 13 nov. 2025.

SBC – SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. **Educação Superior em Computação: dados do Censo da Educação Superior 2017**. Relatório estatístico. São Paulo: Sociedade Brasileira de Computação, 2024. Disponível em: <https://www.sbc.org.br/wp-content/uploads/2024/07/Relatorio-Computac-a-o-INEP-2017.pdf>. Acesso em: 13 nov. 2025.

SCHWENGBER, Jenifer Tais Maciel. **Metodologias ativas e o ensino da computação**. 2024. 40 f. Trabalho de Conclusão de Curso (Licenciatura em Computação e Robótica Educativa) – Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2024.

SEMEEESP – SERVIÇO MANTENEDOR DAS ENTIDADES DO ENSINO SUPERIOR PARTICULAR. **Mapa do Ensino Superior no Brasil 2024**. São Paulo: SEMEESP, 2024. Disponível em: <https://www.semesp.org.br/wp-content/uploads/2024/04/mapa-do-ensino-superior-no-brasil-2024.pdf>. Acesso em: 13 nov. 2025.

SHIMASAKI, Rodrigo. O ensino da programação e o desenvolvimento do pensamento lógico: uma revisão sistemática de literatura. **Revista de Ensino, Educação e Ciências Humanas**, v. 22, p197 - 205, 2021. Disponível em: <https://revistaensinoeducacao.pgskroton.com.br/article/view/8473>. Acesso em: 13 nov. 2025.

SILVA, Cristiane Aparecida da; ROSA, Fabricia Silva da; VICENTE, Ernesto Fernando Rodrigues; ANZILAGO, Marciele. Política de assistência educacional e a taxa de sucesso da graduação das universidades públicas federais. **Revista Brasileira de Política e Administração da Educação**, v. 34, n. 3, p. 837–863, set./dez. 2018. DOI: <https://doi.org/10.21573/vol34n32018.86284>.

SILVA, Danilo Nogueira da. **Lógica de programação: dificuldades de ensino-aprendizagem, métodos e ferramentas computacionais**. 2019. 101 f. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Universidade Estadual de Goiás, Câmpus Anápolis de Ciências Exatas e Tecnológicas Henrique Santillo, Anápolis, 2019. Disponível em: <https://repositorio.ueg.br/jspui/handle/riueg/950>. Acesso em: 10 nov. 2025.

SILVA, Emanuel Leite Oliveira da; FALCÃO, Taciana Pontual. O Pensamento Computacional no Ensino Superior e seu Impacto na Aprendizagem de Programação. Workshop sobre Educação em Computação (WEI), **Anais, 2020**. Disponível em: <https://sol.sbc.org.br/index.php/wei/article/download/11152/11023>. Acesso em: 11 nov. 2025.

SILVA, Polyana Tenório de Freitas e; PINTO, Vannildo Cardoso; BITTENCOURT, Ibsen Mateus; SANTA RITA, Luciana Peixoto. Indicadores de gestão do ensino superior e sua correlação com evasão e retenção: uma análise da taxa de sucesso na graduação em seis instituições federais de ensino entre 2006 e 2015. In: SIMPÓSIO INTERNACIONAL DE GESTÃO DE PROJETOS, INOVAÇÃO E SUSTENTABILIDADE (SINGEP), VI, 2017, São Paulo, SP, Brasil. **Anais [...]**. São Paulo: SINGEP, 2017. Disponível em: <https://singep.org.br/6singep/resultado/506.pdf>. Acesso em: 14 nov. 2025.

SILVA, Priscila da Rosa et al. Fatores relacionados à evasão de estudantes nos cursos de graduação: um estudo no ensino superior brasileiro. **Revista Educar Mais**, Rio Grande, v. 6, n. 3, p. 108–124, 2023. Disponível em: <https://periodicos.ifsul.edu.br/index.php/educarmais/article/view/2970>. Acesso em: 13 nov. 2025.

SILVA, Yuri dos Santos. **O uso de ferramentas computacionais como suporte ao ensino de programação no ensino superior**. 2020. 74 f. Monografia (Graduação em Ciência da Computação) – Universidade Federal de Ouro Preto, Ouro Preto, 2020. Disponível em: [https://monografias.ufop.br/bitstream/35400000/2274/1/MONOGRAFIA\\_UsoFerramentasComputacionais.pdf](https://monografias.ufop.br/bitstream/35400000/2274/1/MONOGRAFIA_UsoFerramentasComputacionais.pdf). Acesso em: 13 nov. 2025.

SOUSA, José de Arimatea de; DO NASCIMENTO, João Carlos Hipólito Bernardes. **Analisando a trajetória do Programa de Apoio a Planos de Reestruturação e Expansão das Universidades Federais (REUNI) na Universidade Federal do Maranhão – Campus Codó (2010-2015)**. Revista Húmus, v. 8, n. 23, 2018. Disponível em: <https://cajapio.ufma.br/index.php/revistahumus/article/view/9485>. Acesso em: 13 nov. 2025.

SOUZA, Andriw Santos de; LIMA, Larissa Corsino; TRINDADE, Genarde Macedo. **Jogo de tabuleiro para ensino de computação por meio de estratégias de solução de problemas**. **Revista Ibero-Americana de Humanidades, Ciências e Educação (REASE)**, São Paulo, v. 11, n. 10, p. 1342–1356, out. 2025.

TEODORO, Leonardo de Almeida; KAPPEL, Marco André Abud. **Aplicação de técnicas de aprendizado de máquina para predição de risco de evasão escolar em instituições públicas de ensino superior no Brasil**. **Revista Brasileira de Informática na Educação**, Porto Alegre, v. 28, p. 838–863, 2020. DOI: 10.5753/rbie.2020.28.0.838. Disponível em: <https://journals-sol.sbc.org.br/index.php/rbie/article/view/3691>. Acesso em: 13 nov. 2025.

UNIVERSIDADE FEDERAL DO MARANHÃO. **Projeto Pedagógico do Curso de Bacharelado Interdisciplinar em Ciência e Tecnologia (BICT)**. São Luís: UFMA, 2016.

UNIVERSIDADE FEDERAL DO MARANHÃO. **Projeto Pedagógico do Curso de Engenharia da Computação**. São Luís: UFMA, 2014.

VIDENOVİK, Maja; VOLD, Tone; KIØNIG, Linda; MADEVSKA BOGDANOVA, Ana; TRAJKOVİK, Vladimir. **Game-based learning in computer science education: a scoping literature review**. **International Journal of STEM Education**, v. 10, n. 1, p. 54, 06 set. 2023. DOI: <https://doi.org/10.1186/s40594-023-00447-2>. Disponível em: <https://www.springer.com/journal/40594/volumes-and-issues/10-1>. Acesso em: 13 nov. 2025.

## APÊNDICE A – TABELA DAS QUESTÕES UTILIZADAS NO JOGO

FASE 1		
PERGUNTAS	RESPOSTAS	REFERÊNCIAS
O que é um algoritmo?	Um conjunto finito de passos bem definidos para resolver um problema	<p>MANZANO, José Augusto Navarro Garcia. <b>Algoritmos: lógica para desenvolvimento de programação de computadores</b>. 29. ed. São Paulo: Érica, 2010.</p> <p>CORMEN, Thomas H. <i>et al.</i> <b>Algoritmos: teoria e prática</b>. 3. ed. Rio de Janeiro: Elsevier, 2012.</p>
Qual característica é essencial em um algoritmo?	Ter passos bem definidos	
Um algoritmo deve ser:	Finito	
Qual é o objetivo principal de um algoritmo?	Resolver um problema	
Qual das opções NÃO é uma característica de algoritmos?	Ambiguidade	
Em programação, para que servem as variáveis?	Armazenar valores temporários	
Qual situação representa melhor a entrada de dados em um programa?	Digitar informações pelo teclado	
O que significa dizer que um programa segue uma ordem lógica?	As instruções são executadas em sequência organizada	
Qual alternativa representa um exemplo de processamento?	Calcular a média de notas	
Qual das opções NÃO faz parte de um programa simples?	Compilador	
Um algoritmo que nunca termina viola qual característica?	Finitude	
Dois algoritmos resolvem o mesmo problema corretamente. Qual critério pode diferenciá-los?	A eficiência	
Um algoritmo correto é aquele que	Sempre retorna o resultado esperado	
Qual fator NÃO influencia a eficiência de um algoritmo?	Cor do código	
Quando dizemos que um algoritmo é genérico, significa que ele:	Resolve uma classe de problemas	
Um programa que soma dois números precisa obrigatoriamente:	Ler os valores e realizar o cálculo	
Qual situação exige o uso de uma decisão no programa?	Verificar se um valor é maior que outro	
Um programa que executa instruções sempre na mesma ordem é chamado de	Sequencial	
Um erro de lógica ocorre quando:	O programa executa, mas gera resultado errado	
Qual é o principal objetivo de organizar bem a lógica de um programa?	Facilitar manutenção e entendimento	
Pergunta: Um algoritmo correto, porém infinito, é considerado:	Incorreto	
Por que algoritmos devem ser independentes de linguagem?	Para facilitar análise e reutilização	
Ao comparar algoritmos, qual critério deve ser analisado primeiro?	Correção	
A principal contribuição do estudo de algoritmos é:	Desenvolver raciocínio lógico para resolver problemas	
Ao desenvolver um programa, o programador decide verificar se os dados digitados são válidos antes de realizar qualquer cálculo. Essa decisão melhora principalmente:	A confiabilidade dos resultados	
Um programa apresenta resultados corretos na maioria dos casos, mas falha em situações específicas. Esse tipo de erro é mais difícil de identificar porque:	O erro não aparece sempre	

Um programa executa corretamente, mas realiza cálculos desnecessários que não influenciam no resultado final. Esse problema está relacionado a:	Falta de eficiência	
Um algoritmo (ou lógica de solução) que só funciona para um conjunto muito restrito de dados apresenta qual limitação conceitual?	Falta de generalização	
Antes mesmo de escrever qualquer código, um programador desenha o fluxo das decisões e ações do programa. Essa prática contribui principalmente para:	Reduzir erros de lógica	
<b>FASE 2</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
Qual é o principal objetivo da Engenharia de Software?	Desenvolver software de forma organizada, confiável e sustentável	<p>DATE, C. J. <b>Introdução a sistemas de bancos de dados</b>. Tradução de Daniel Vieira. 8. ed. Rio de Janeiro: Elsevier, 2003.</p> <p>SEBESTA, Robert W. <b>Conceitos de linguagens de programação</b>. 9. ed. Porto Alegre: Bookman, 2011.</p> <p>SOMMERVILLE, Ian. <b>Engenharia de software</b>. 9. ed. São Paulo: Pearson Prentice Hall, 2011.</p>
Um requisito descreve:	O que o software deve fazer	
Um banco de dados serve principalmente para:	Organizar informações	
O software que gerencia um banco de dados é chamado de:	SGBD	
Integridade de dados está relacionada a:	Consistência	
Qual paradigma prioriza a execução de comandos em sequência explícita?	Imperativo	
No paradigma orientado a objetos, o foco principal está em	Objetos e interações	
Qual paradigma descreve o que deve ser feito, não como?	Declarativo	
Por que softwares precisam de manutenção?	Porque requisitos e contextos mudam ao longo do tempo	
“Consulta” em banco de dados é:	Busca de dados	
Um software funciona corretamente, mas é difícil de modificar. Qual qualidade está comprometida?	Manutenibilidade	
Por que mudanças de requisitos são comuns em projetos de software?	O ambiente e as necessidades evoluem	
Qual prática ajuda a reduzir riscos em projetos de software?	Planejamento e acompanhamento contínuo	
Redundância excessiva tende a causar:	Inconsistência	
Independência de dados significa:	Dados menos acoplados	
Concorrência em banco de dados trata de:	Vários acessos	
Ao evitar variáveis globais e estados mutáveis, um programador se aproxima de qual paradigma?	Funcional	
Encapsulamento está mais associado a:	Programação orientada a objetos	
Qual paradigma favorece paralelismo devido à ausência de efeitos colaterais?	Funcional	
Qual fator mais contribui para o fracasso de projetos de software?	Falhas de comunicação e gestão	
Um software atende todos os requisitos definidos, mas os usuários estão insatisfeitos. Qual falha é mais provável?	Requisitos mal compreendidos	
Por que projetos de software são considerados inerentemente complexos?	Porque lidam com problemas abstratos e mutáveis	
Um sistema apresenta dados diferentes para o mesmo cliente dependendo do relatório consultado. Qual é a causa mais provável?	Redundância mal controlada	
Dois usuários atualizam o mesmo registro quase simultaneamente e uma alteração é perdida. Qual conceito não foi tratado adequadamente?	Concorrência	

Após uma falha de energia, parte das atualizações foi salva e parte não. Qual princípio foi violado?	Atomicidade	
Um sistema apresenta muitos erros causados por alterações inesperadas de estado. Qual paradigma tende a reduzir esse problema?	Funcional	
Ao escolher orientação a objetos, o maior risco mal gerenciado é:	Acoplamento excessivo	
Um software tecnicamente correto, mas difícil de usar, falha principalmente em qual aspecto?	Usabilidade	
Em projetos grandes, por que dividir o sistema em partes menores é essencial?	Para reduzir a complexidade e facilitar gestão	
A principal contribuição dos paradigmas para a programação é:	Organizar formas de pensar soluções	
<b>FASE 3</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
Qual conjunto representa corretamente o sistema binário?	0 e 1	<p>BOYLESTAD, Robert L.; NASHELSKY, Louis. <b>Dispositivos eletrônicos e teoria de circuitos</b>. 8. ed. São Paulo: Pearson Prentice Hall, 2004.</p> <p>EDMINISTER, Joseph A. <b>Circuitos elétricos</b>. 2. ed. São Paulo: Pearson Education do Brasil, 1991. (Coleção Schaum).</p> <p>TOCCI, Ronald J.; WIDMER, Neal S.; MOSS, Gregory L. <b>Sistemas digitais: princípios e aplicações</b>. 11. ed. São Paulo: Pearson, 2011.</p>
A unidade de medida da corrente elétrica é:	Ampère	
Um circuito retificador é usado para:	Converter CA em CC	
Um botão gera múltiplos pulsos indesejados ao ser pressionado. O fenômeno observado é:	Bounce	
Um fusível queima logo ao ligar o circuito. O problema mais provável é:	Curto-circuito	
A principal função de uma porta NOT é:	Inverter o nível lógico	
Um resistor é usado principalmente para:	Limitar corrente	
A potência elétrica é calculada por:	$P = V \cdot I$	
Ao energizar um CI digital, nenhuma saída responde. O primeiro ponto a conferir é:	Conexão de GND e VCC	
Ao medir continuidade, o multímetro indica valor infinito. Isso significa:	Circuito aberto	
Um flip-flop é usado principalmente para:	Armazenar informação	
Em um circuito série, a corrente é:	Igual em todos os pontos	
Um transistor em saturação está operando como:	Chave fechada	
Em um contador, a contagem “pula” estados. A causa típica é:	Atraso de propagação	
Um transistor não entra em condução. O primeiro ponto a verificar é:	Polarização da base	
Qual porta lógica é considerada universal?	NAND	
Um capacitor em série com um sinal CA tende a:	Bloquear CC	
Ao medir corrente, o instrumento deve ser ligado:	Em série	
Capacitores próximos ao CI servem principalmente para:	Desacoplar alimentação	
Após retificação, a tensão ainda apresenta ondulação. Qual componente reduz isso?	Capacitor	
Um amplificador apresenta corte do sinal nas extremidades. A causa mais provável é:	Polarização incorreta	
Em um sistema síncrono, erros de temporização estão ligados a:	Atrasos de propagação	
Um flip-flop entra em estado imprevisível ao ligar. O fenômeno associado é:	Metastabilidade	
Dois resistores idênticos em paralelo resultam em uma resistência equivalente:	Metade de um resistor	
Um circuito apresenta comportamento diferente no simulador e no laboratório. A principal causa é:	Componentes reais não ideais	

Um erro de “glitch” em circuitos digitais ocorre devido a:	Diferença de atrasos	
Se a tensão de saída varia com a carga, o circuito apresenta:	Má regulação	
Uma instalação apresenta queda excessiva de tensão sob carga. A causa mais provável é:	Alta resistência dos condutores	
Um sistema síncrono apresenta falhas apenas em alta frequência. O limite foi imposto por:	Atraso de propagação	
O osciloscópio mostra ruído excessivo na saída. A melhor correção é:	Melhorar aterramento	
<b>FASE 4</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
(Projeto e Desenvolvimento) O principal objetivo da fase de análise de sistemas é:	Entender o problema	
(IA) Um agente inteligente é definido como uma entidade que:	Interage com um ambiente	
(Lógica/Matemática Discreta) Uma proposição é:	Uma frase com valor verdadeiro ou falso	
(Redes) Qual dispositivo encaminha pacotes entre redes diferentes?	Roteador	GERSTING, Judith L. <b>Fundamentos matemáticos para a ciência da computação:</b> matemática discreta e suas aplicações. Rio de Janeiro: LTC, 2010.
(Projeto e Desenvolvimento) Um requisito funcional descreve:	O comportamento do sistema	
(IA) Um sistema especialista é baseado principalmente em:	Regras e conhecimento	
Uma tabela-verdade serve para:	Testar proposições lógicas	
(Redes) Um protocolo de rede define:	Regras de comunicação	
(Projeto e Desenvolvimento) Casos de uso servem para representar:	Interações com usuários	
(IA) A IA simbólica trabalha principalmente com:	Símbolos e regras	KUROSE, James F.; ROSS, Keith W. <b>Redes de computadores e a internet:</b> uma abordagem top-down. 6. ed. São Paulo: Pearson Education do Brasil, 2013.
(Projeto e Desenvolvimento) Um problema comum quando requisitos são mal definidos é:	Retrabalho	
(IA) Uma heurística é usada para:	Estimar custo de solução	
(Lógica/Matemática Discreta) Uma relação de equivalência deve ser:	Reflexiva, simétrica e transitiva	
(Redes) O UDP é mais adequado quando a aplicação prioriza:	Baixa latência	
(Projeto e Desenvolvimento) Separar responsabilidades no sistema contribui para:	Melhor manutenção	
(IA) O aprendizado supervisionado depende de:	Exemplos rotulados	
(Lógica/Matemática Discreta) A proposição “Se p então q” é falsa quando:	p é verdadeiro e q é falso	
A principal função do DNS é:	Traduzir nomes em endereços	
(Projeto e Desenvolvimento) No projeto orientado a objetos, alta coesão significa:	Responsabilidades relacionadas	
(IA) Em métodos de busca, o espaço de estados representa:	Todas as situações possíveis	
(Projeto e Desenvolvimento) Quando requisitos mudam frequentemente, a melhor estratégia de projeto é priorizar	Modularidade	
(IA) Qual abordagem de IA lida melhor com incerteza explícita?	Métodos probabilísticos	
(Lógica/Matemática Discreta) A negação correta de “Todos os elementos têm propriedade P” é:	Existe um elemento sem P	
(Redes) O desempenho percebido pelo usuário depende mais de:	Atraso e perda de pacotes	
(Projeto e Desenvolvimento) Um sistema que cresce sem revisão de projeto tende a:	Aumentar complexidade	WAZLAWICK, Raul Sidnei. <b>Análise e projeto de sistemas de informação orientados a objetos.</b> Rio de Janeiro: Elsevier, 2004.

(IA) Redes neurais são menos interpretáveis porque:	Aprendem pesos distribuídos	
(Lógica/Matemática Discreta) Em lógica proposicional, um conjunto inconsistente de premissas:	Permite qualquer conclusão	
Uma rede pode apresentar alta largura de banda e ainda assim ser lenta devido a:	Alta latência	
(Projeto e Desenvolvimento) Quando diferentes equipes trabalham no mesmo sistema, o maior risco é:	Falta de alinhamento conceitual	
(Redes) O uso de buffers excessivos em roteadores pode causar:	Bufferbloat	
<b>FASE 5</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
Arquitetura) O componente responsável por executar instruções é:	Unidade Central de Processamento	
(Sistemas Operacionais) Um processo pode ser definido como:	Um programa em execução	
(Instrumentação) Um transdutor realiza a conversão de:	Grandeza física em sinal	
(Sistemas Lineares) Um sistema linear satisfaz o princípio da:	Superposição	BALBINOT, Alexandre;
(Arquitetura) O clock do sistema define principalmente:	Frequência de operação	BRUSAMARELLO, Valner João.
(Sistemas Operacionais) O kernel de um sistema operacional é responsável por:	Gerenciamento central do sistema	<b>Instrumentação e fundamentos de medidas.</b> 2. ed. Rio de Janeiro: LTC, 2010. v. 1.
(Instrumentação) Instrumentos analógicos apresentam saída:	Contínua	
(Sistemas Lineares) Um sistema causal responde apenas a entradas:	Presentes e passadas	
A ULA (Unidade Lógica e Aritmética) realiza:	Operações aritméticas e lógicas	GEROMEL, José C.;
(Instrumentação) Instrumentação está relacionada principalmente à:	Medição	KOROGUI, Rubens H. <b>Controle linear de sistemas dinâmicos:</b> teoria, ensaios práticos e exercícios. São Paulo: Blucher, 2011.
(Arquitetura) O pipeline melhora desempenho ao:	Executar instruções em paralelo parcial	
(Sistemas Operacionais) Deadlock ocorre quando processos:	Esperam indefinidamente por recursos	
(Instrumentação) A precisão está relacionada à:	Repetibilidade	
(Sistemas Lineares) Polos com parte real positiva indicam sistema:	Instável	TANENBAUM, Andrew S.;
(Arquitetura) A cache existe para:	Reduzir latência de acesso	AUSTIN, Todd. <b>Arquitetura e organização de computadores.</b> 5. ed. São Paulo: Pearson Prentice Hall, 2010.
(Sistemas Operacionais) A paginação é usada para:	Gerenciar memória	
(Instrumentação) A calibração tem como objetivo:	Ajustar leituras	
(Sistemas Lineares) O tempo de acomodação indica:	Tempo até estabilidade	
(Arquitetura) Endereçamento imediato significa que o operando está:	Na própria instrução	
(Sistemas Operacionais) Memória virtual permite:	Executar programas maiores que a RAM	TANENBAUM, Andrew S.;
(Arquitetura) A hierarquia de memória explora o princípio de:	Localidade	BOS, Herbert. <b>Sistemas operacionais modernos.</b> 4. ed. São Paulo: Pearson, 2016.
(Sistemas Operacionais) A troca excessiva de páginas indica:	Thrashing	
(Instrumentação) Um instrumento pode ser preciso e não exato quando:	Mede longe do real	
(Sistemas Lineares) Zeros no semiplano direito indicam sistema:	Não mínimo de fase	
(Arquitetura) Mesmo com processador rápido, um sistema pode ser lento devido a:	Cache ineficiente	

(Sistemas Operacionais) Threads reduzem overhead porque:	Compartilham recursos do processo	
(Instrumentação) O erro conceitual mais grave em instrumentação é confundir:	Precisão e exatidão	
(Sistemas Lineares) O critério de Routh avalia a:	Estabilidade	
(Arquitetura) Se o tempo de acesso à memória domina o ciclo, a melhor melhoria é:	Melhorar cache	
(Sistemas Operacionais) Sem mecanismos de sincronização, sistemas concorrentes podem sofrer de:	Condição de corrida	
<b>FASE 6</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
(Métodos Formais) O principal objetivo dos métodos formais é:	Especificar e verificar sistemas de forma matemática	<p>GONZALEZ, Rafael C.; WOODS, Richard E. <b>Processamento de imagens digitais: fundamentos</b>. 2. ed. São Paulo: Pearson Prentice Hall, 2008.</p> <p>MENEZES, Paulo Blauth. <b>Linguagens formais e autômatos</b>. 6. ed. Porto Alegre: Bookman, 2011.</p> <p>O'REGAN, Gerard. <b>Concise guide to formal methods: theory, fundamentals and industry applications</b>. Cham: Springer, 2017.</p> <p>TANENBAUM, Andrew S.; VAN STEEN, Maarten. <b>Sistemas distribuídos: princípios e paradigmas</b>. 2. ed. São Paulo: Pearson Prentice Hall, 2007.</p>
(Computação Gráfica) Uma imagem digital é formada por uma matriz de:	Pixels	
(Sistemas Distribuídos) A exclusão mútua distribuída garante que:	Apenas um processo acesse recurso	
(Linguagens Formais e Autômatos) Expressões regulares descrevem:	Linguagens regulares	
(Métodos Formais) Uma especificação formal é caracterizada por ser:	Precisa e não ambígua	
(Computação Gráfica) A digitalização de imagens envolve:	Amostragem e quantização	
(Linguagens Formais e Autômatos) Uma linguagem formal é definida como:	Um conjunto de palavras	
(Sistemas Distribuídos) A comunicação em sistemas distribuídos ocorre por troca de:	Mensagens	
(Métodos Formais) Um erro comum evitado por métodos formais é:	Ambiguidade nos requisitos	
(Computação Gráfica) O histograma de uma imagem descreve a:	Distribuição de intensidades	
(Métodos Formais) Um modelo inconsistente indica que:	Existem estados contraditórios	
Um filtro passa-baixa é utilizado para:	Suavizar variações	
(Sistemas Distribuídos) A latência de rede afeta diretamente:	Tempo de resposta	
(Linguagens Formais e Autômatos) A hierarquia de Chomsky classifica linguagens por:	Complexidade	
(Métodos Formais) Refinar uma especificação significa:	Torná-la mais detalhada preservando o significado	
(Computação Gráfica) A amostragem espacial define a:	Densidade de pixels	
(Sistemas Distribuídos) A exclusão mútua distribuída garante que:	Apenas um processo acesse recurso	
(Linguagens Formais e Autômatos) Uma gramática é definida por:	Variáveis, terminais e regras	
(Métodos Formais) Modelar falhas explicitamente permite:	Analisar comportamentos não ideais	
(Computação Gráfica) A filtragem espacial depende principalmente de:	Kernel utilizado	
(Métodos Formais) Um sistema pode ser considerado correto quando:	Satisfaz formalmente sua especificação	
(Computação Gráfica) A transformada de Fourier permite analisar o conteúdo no:	Domínio da frequência	
(Sistemas Distribuídos) O trade-off CAP afirma que não é possível garantir:	Consistência, disponibilidade e tolerância a falhas	

(Linguagens Formais e Autômatos) Linguagens regulares são fechadas sob:	Interseção, Complemento, União	
(Métodos Formais) O principal limite do model checking é:	Explosão do espaço de estados	
(Computação Gráfica) O aliasing ocorre quando a amostragem é:	Insuficiente	
(Sistemas Distribuídos) A principal dificuldade do consenso distribuído é lidar com:	Falhas	
(Linguagens Formais e Autômatos) Nem toda linguagem livre de contexto é:	Regular	
(Métodos Formais) Uma propriedade de vivacidade garante que:	Algo desejado ocorrerá eventualmente	
(Computação Gráfica) Um erro conceitual comum é confundir:	Amostragem e quantização	
<b>FASE 7</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
(Processamento Digital de Sinais) A frequência de amostragem é medida em:	Hertz	DINIZ, Paulo S. R.; SILVA, Eduardo A. B. da; NETTO, Sergio L. <b>Processamento digital de sinais:</b> projeto e análise de sistemas. 2. ed. Porto Alegre: Bookman, 2014.  OGATA, Katsuhiko. <b>Engenharia de controle moderno.</b> 5. ed. São Paulo: Pearson Prentice Hall, 2010.
(Engenharia de Controle) Um sistema em malha aberta não utiliza:	Sensor	
(Processamento Digital de Sinais) Um sinal digital é definido em:	Tempo discreto	
(Engenharia de Controle) A função de transferência relaciona:	Entrada e saída	
(Processamento Digital de Sinais) A quantização está associada à:	Amplitude	
(Engenharia de Controle) O erro é a diferença entre:	Referência e saída	
(Processamento Digital de Sinais) Um sistema digital processa sinais:	Contínuos	
(Engenharia de Controle) Um sistema linear satisfaz o princípio da:	Superposição	
(Processamento Digital de Sinais) Um sinal causal é não nulo para:	$n \geq 0$	
(Engenharia de Controle) Um sistema de controle tem como objetivo:	Regular variáveis	
(Engenharia de Controle) O erro em regime permanente depende do:	Tipo do sistema	
(Processamento Digital de Sinais) A transformada Z é usada para análise no domínio:	Da frequência complexa	
(Engenharia de Controle) A estabilidade de um sistema contínuo depende dos:	Polos	
(Processamento Digital de Sinais) A convolução descreve a relação entre:	Entrada e saída	
(Engenharia de Controle) Um sistema subamortecido apresenta:	Oscilações	
(Processamento Digital de Sinais) Um sistema FIR é caracterizado por resposta ao impulso:	Finita	
(Engenharia de Controle) A resposta ao degrau é usada para avaliar:	Desempenho	
(Processamento Digital de Sinais) O aliasing ocorre quando:	Há ruído térmico	
(Engenharia de Controle) A realimentação negativa tende a:	Melhorar estabilidade	
(Processamento Digital de Sinais) A invariância no tempo significa que:	Um atraso gera atraso	
(Processamento Digital de Sinais) Um sistema causal exige ROC:	À direita do polo	

(Engenharia de Controle) O critério de Routh-Hurwitz avalia a:	Estabilidade	
(Processamento Digital de Sinais) O teorema da amostragem exige $f_s$ maior que:	$2f_{max}$	
(Engenharia de Controle) Zeros no semiplano direito caracterizam sistema:	Não mínimo de fase	
(Processamento Digital de Sinais) Um sistema IIR possui resposta ao impulso:	Infinita	
(Engenharia de Controle) O ganho excessivo pode causar:	Instabilidade	
(Processamento Digital de Sinais) A FFT é um algoritmo para calcular:	DFT	
(Engenharia de Controle) Um controlador proporcional afeta diretamente o:	Tempo de resposta	
(Processamento Digital de Sinais) Um filtro ideal não é realizável porque:	Não é causal	
(Engenharia de Controle) Um sistema pode ser estável e apresentar:	Oscilações transitórias	
<b>FASE 8</b>		
<b>PERGUNTAS</b>	<b>RESPOSTAS</b>	<b>REFERÊNCIAS</b>
(Compiladores) Um compilador tem como função principal:	Traduzir código	<p>ALMEIDA, Rodrigo Maximiano Antunes de; MORAES, Carlos Henrique Valério de; SERAPHIM, Thatyana de Faria Piola. <b>Programação de sistemas embarcados:</b> desenvolvendo software para microcontroladores em linguagem C. Rio de Janeiro: Elsevier, 2016.</p> <p>COOPER, Keith D.; TORCZON, Linda. <b>Construindo compiladores.</b> Porto Alegre: Bookman, 2013.</p> <p>PETROU, Maria; PETROU, Costas. <b>Image processing: the fundamentals.</b> 2. ed. Chichester: John Wiley &amp; Sons, 2010.</p> <p>PRUDENTE, Francesco. <b>Automação industrial PLC:</b> teoria e aplicações. 2. ed. Rio de Janeiro: LTC, 2011.</p>
(Sistemas Embarcados) Um microcontrolador integra em um único chip:	CPU e periféricos	
(Automação Industrial) Um processo automatizado é aquele que:	Opera sem intervenção constante	
(Processamento de Imagem) Uma imagem binária possui pixels com valores:	0 e 1	
(Compiladores) A análise léxica identifica:	Tokens	
(Sistemas Embarcados) O firmware é armazenado normalmente em:	Flash	
(Processamento de Imagem) O histograma de uma imagem descreve a:	Distribuição de intensidades	
(Automação Industrial) Um sistema automatizado básico contém:	Sensor, controlador e atuador	
(Compiladores) A análise sintática verifica:	Estrutura gramatical	
(Sistemas Embarcados) Sistemas embarcados costumam operar com:	Recursos limitados	
(Compiladores) A tabela de símbolos armazena principalmente:	Identificadores	
(Sistemas Embarcados) Em sistemas de tempo real, perder um deadline pode causar:	Falha do sistema	
(Automação Industrial) O ciclo de varredura de um CLP envolve:	Entrada, execução e saída	
(Processamento de Imagem) O processamento no domínio espacial atua diretamente nos:	Pixels	
(Compiladores) O IR (código intermediário) representa o programa de forma:	Independente de máquina	
(Sistemas Embarcados) Interrupções são usadas para:	Responder a eventos	
(Automação Industrial) A temporização em CLPs é realizada por:	Temporizadores	
(Processamento de Imagem) A convolução em imagens envolve:	Máscara e vizinhança	
(Compiladores) A eliminação de código morto remove:	Instruções sem efeito	
(Sistemas Embarcados) O watchdog timer é usado para:	Reiniciar em falhas	

(Compiladores) Um compilador otimiza sem alterar:	A semântica	
(Sistemas Embarcados) Um sistema hard real-time exige que deadlines:	Nunca sejam perdidos	
(Automação Industrial) Em sistemas industriais, falhas devem ser tratadas com foco em:	Segurança	
(Processamento de Imagem) A transformada de Fourier em imagens analisa o:	Domínio da frequência	
(Compiladores) A SSA facilita principalmente:	Análise de dados	
(Sistemas Embarcados) O uso inadequado de delays pode provocar:	Bloqueio do sistema	
(Automação Industrial) Controle em malha fechada utiliza:	Realimentação	
(Processamento de Imagem) Altas frequências em imagens correspondem a:	Bordas e detalhes	
(Compiladores) A alocação ingênua de registradores causa:	Spill excessivo	
(Processamento de Imagem) A segmentação baseada em limiarização depende principalmente da:	Intensidade	

**APÊNDICE B – TABELA DOS PERSONAGENS E SUAS HABILIDADES**

<b>Personagem</b>	<b>Habilidade</b>
Calouro Interessado	Uma única vez na partida, pode tratar uma carta Difícil como carta Média.
Preguiçoso	Uma única vez na partida, pode ignorar completamente os efeitos negativos de uma carta Difícil, mas perde a próxima rodada.
Monitora solidária	Uma única vez na partida, ao comprar uma carta Difícil, compra duas, é realizada a leitura da carta, escolhe uma para responder e avança uma casa
Veterano	Uma única vez na partida, pode ignorar completamente o efeito de uma carta (incluindo carta VIDA)
Programador Pragmático	Uma única vez na partida, ao receber um efeito negativo, pode transferi-lo para outro jogador à sua escolha.
Teórica Abstrata	Uma única vez na partida, ao comprar uma carta Difícil, pode tratá-la como carta Média e ainda avançar 1 casa no tabuleiro.

## APÊNDICE C – QUESTIONÁRIO DE AVALIAÇÃO DO JOGO

# AVALIAÇÃO COMPILA



Perguntas	Discordo Totalmente	Discordo	Neutro	Concordo	Concordo Totalmente
1. Eu acho que gostaria de usar essa aplicação com frequência	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. Eu acho o sistema desnecessariamente complexo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Eu achei o sistema fácil de usar	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Eu acho que as várias funções do sistema estão muito bem integradas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

# AVALIAÇÃO COMPILA



Perguntas	Discordo Totalmente	Discordo	Neutro	Concordo	Concordo Totalmente
6. Eu acho que o sistema apresenta muitas inconsistências	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7. Eu imagino que as pessoas aprenderão a usar esse sistema rapidamente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8. Eu achei o sistema complicado de usar	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9. Eu me senti confiante ao usar o sistema	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10. Eu precisei aprender várias coisas novas antes de usar o sistema	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**APÊNDICE D – TABELA DE RESPOSTAS INDIVIDUAIS DE AVALIAÇÃO DO SUS**

		NÚMERO DAS PERGUNTAS									
		1	2	3	4	5	6	7	8	9	10
Nº DE PARTICIPANTES	1	4	1	5	1	4	3	5	1	5	1
	2	4	1	2	1	5	3	5	3	3	2
	3	5	1	5	1	4	3	5	1	5	1
	4	5	1	4	1	3	2	5	1	5	1
	5	4	2	3	1	4	2	5	1	4	1
	6	4	1	4	1	4	2	5	2	5	1