



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS VII GOVERNADOR ANTÔNIO MARIZ
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

WENDELL FERREIRA DE ANDRADE

**VERIFICAÇÃO DE OBJETOS DE APRENDIZAGEM NO COMPONENTE
CURRICULAR DA DISCIPLINA DE ESTRUTURA DE DADOS**

**PATOS - PB
2023**

WENDELL FERREIRA DE ANDRADE

**VERIFICAÇÃO DE OBJETOS DE APRENDIZAGEM NO COMPONENTE
CURRICULAR DA DISCIPLINA DE ESTRUTURA DE DADOS**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Universidade Estadual da Paraíba, Campus VII, em cumprimento à exigência para obtenção do título de Bacharel em Ciência da Computação.

Orientadora: Profa. Ma. Keila Lucas dos Santos

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

A554v Andrade, Wendell Ferreira de.
Verificação de objetos de aprendizagem no componente curricular da disciplina de estrutura de dados [manuscrito] / Wendell Ferreira de Andrade. - 2023.
43 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas, 2023.

"Orientação : Profa. Ma. Keila Lucas dos Santos, Coordenação do Curso de Computação - CCEA. "

1. Ensino da programação. 2. Objetos de aprendizagem. 3. Estrutura de dados. I. Título

21. ed. CDD 005.3

WENDELL FERREIRA DE ANDRADE

**VERIFICAÇÃO DE OBJETOS DE APRENDIZAGEM NO COMPONENTE
CURRICULAR DA DISCIPLINA DE ESTRUTURA DE DADOS**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Universidade Estadual da Paraíba, Campus VII, em cumprimento à exigência para obtenção do título de Bacharel em Ciência da Computação.

Orientadora: Profa. Ma. Keila Lucas dos Santos

Aprovado em 30/11/2023

BANCA EXAMINADORA

Keila Lucas dos Santos

Profa. Msc. Keila Lucas dos Santos
(Orientadora)

Angélica Felix Medeiros

Profa. Msc. Angélica Felix Medeiros
(Examinadora)

Janderson Jason Barbosa Aguiar

Prof. Dr. Janderson Jason Barbosa Aguiar
(Examinador)

RESUMO

Na área da tecnologia, especialmente nos cursos de Computação, é possível observar diversos desafios que impactam diretamente os estudantes e o mercado de trabalho. Um dos principais problemas enfrentados é o alto índice de evasão nos cursos de Computação. Muitos alunos desistem ao longo do caminho devido às dificuldades de aprendizagem e às complexidades inerentes aos conteúdos abordados. Para superar esses desafios, é necessário investir em estratégias que auxiliem os alunos no processo de aprendizagem, como o uso de recursos educacionais digitais, tutoriais e metodologias inovadoras de ensino. Este trabalho tem como objetivo aplicar e validar uma metodologia ativa com base em Objetos de Aprendizagem (OA) no componente curricular da disciplina de Estrutura de Dados. Busca-se analisar a eficiência desses OA na disciplina e explorar possíveis soluções para melhorar o processo de ensino e aprendizado. Além disso, pretende-se compreender de forma mais precisa e abrangente as dificuldades enfrentadas pelos estudantes no processo de aprendizagem da Programação, incluindo a identificação de obstáculos comuns e a análise de soluções específicas para tornar o ensino de Programação mais acessível e eficaz.

Palavras-chave: Ensino da Programação; Aprendizagem Efetiva; Objetos de Aprendizagem; Estrutura de Dados; Listas Simplesmente Encadeadas.

ABSTRACT

In the technology field, especially in Computer Science courses, various challenges impacting students and the job market can be observed. One of the primary issues faced is the high dropout rate in Computer Science programs. Many students abandon the path due to learning difficulties and the inherent complexities of the covered content. To overcome these challenges, it is necessary to invest in strategies that assist students in the learning process, such as the use of digital educational resources, tutorials, and innovative teaching methodologies. This work aims to apply and validate an active methodology based on Learning Objects (LO) in the curricular component of the Data Structures discipline. The goal is to analyze the efficiency of these LO in the discipline and explore possible solutions to improve the teaching and learning process. Additionally, there is an intention to understand more precisely and comprehensively the difficulties faced by students in the Programming learning process, including the identification of common obstacles and the analysis of specific solutions to make Programming education more accessible and effective.

Keywords: Teaching Programming; Effective Learning; Learning Objects; Data Structures; Singly Linked Lists.

LISTA ABREVIATURAS E SIGLAS

| | |
|-----|-----------------------------------|
| CS | COMPUTER SCIENCE |
| DS | DESIGN SCIENCE |
| EDS | ESTRUTURA DE DADOS |
| OA | OBJETO DE APRENDIZAGEM |
| PC | PENSAMENTO COMPUTACIONAL |
| RSL | REVISÃO SISTEMÁTICA DE LITERATURA |
| TI | TECNOLOGIA DA INFORMAÇÃO |

SUMÁRIO

| | |
|-----------------------------------------------------------------|-----------|
| 1. INTRODUÇÃO | 9 |
| 1.1. Objetivo Geral..... | 10 |
| 1.2. Objetivo Específicos..... | 11 |
| 2. REFERENCIAL TEÓRICO | 12 |
| 2.1. Educação e programação em Computação..... | 12 |
| 2.2. Estrutura de Dados..... | 14 |
| 2.2.1. Estrutura de Listas..... | 15 |
| 2.3. Objetos de Aprendizagem | 16 |
| 2.4. Computação Desplugada..... | 18 |
| 3. TRABALHOS RELACIONADOS..... | 19 |
| 4. METODOLOGIA..... | 22 |
| 4.1. Revisão Sistemática de Literatura | 22 |
| 4.1.1. Planejamento da RSL..... | 23 |
| 4.1.2. Desenvolvimento da RSL..... | 23 |
| 4.2. Design do Experimento | 24 |
| 4.2.1. Planejamento do Experimento | 25 |
| 4.2.2. Estrutura do Objeto de Aprendizagem..... | 25 |
| 4.2.3. Participantes do Experimento..... | 28 |
| 4.2.4. Execução do Experimento | 29 |
| 5. RESULTADOS | 31 |
| 5.1. Resultado da RSL | 31 |
| 5.2. Questões da Pesquisa | 32 |
| 5.1. Resultados do Experimento..... | 33 |
| 6. CONCLUSÃO..... | 37 |
| REFERÊNCIAS | 39 |
| APÊNDICE A - PLANO DE AULA | 42 |
| APÊNDICE B - QUESTIONÁRIO UTILIZADO NO EXPERIMENTO | 44 |

1. INTRODUÇÃO

A aprendizagem da Programação é uma área de estudo muito importante para o desenvolvimento tecnológico, impulsionada pelos avanços que transformam a forma de viver, trabalhar e se comunicar (Campbell-Kelly *et al.*, 2014). No entanto, apesar do seu impacto significativo, tanto os alunos como os professores enfrentam desafios no processo de ensino e aprendizagem da Programação.

Uma das principais dificuldades enfrentadas pelos alunos no estudo da programação é a abstração dos conteúdos das disciplinas de Algoritmos e Estruturas de Dados (EDS). Conceitos de Algoritmos e EDS podem ser abstratos e complexos de compreender. A lógica e o pensamento exigidos pela programação podem ser diferentes daqueles usados em outras disciplinas, como *Matemática* e *Linguística*, o que pode dificultar a assimilação dos conceitos por parte dos alunos (Smith; Johnson, 2018). A natureza dinâmica da programação exige dos alunos adaptação e aprendizado contínuo.

Por outro lado, os professores de programação também enfrentam desafios ao transmitir o conhecimento de forma eficiente aos alunos. A velocidade com que novas tecnologias e tendências surgem na programação também representa um desafio para os professores, que precisam estar constantemente atualizados para oferecer um ensino relevante e de qualidade (Brown; Ranellucci, 2019). Essas dificuldades enfrentadas tanto pelos alunos quanto pelos professores podem resultar em altos índices de evasão nos cursos de programação. Muitos estudantes desistem do curso devido a diversos fatores, entre eles à dificuldade de acompanhar o ritmo acelerado das disciplinas, à falta de motivação e interesse, à falta de apoio adequado ou à sensação de que a área não é adequada para eles (Guzdial, 2017).

As falhas no aprendizado na área de Programação, influencia o número de pessoas bem qualificadas e isso gera um impacto negativo no mercado de trabalho. Com o avanço tecnológico constante e a crescente demanda por profissionais qualificados em programação, a escassez de graduados com boas habilidades em computação cria um déficit empregatício, e muitas vagas de trabalho ficam em aberto, em diversas áreas da Tecnologia da Informação (TI). Essa situação é ressaltada por um estudo realizado pela UNESCO em 2018, que apontou a escassez global de profissionais de TI, destacando a necessidade urgente de investir em programas de educação em programação para suprir a demanda do mercado.

Para enfrentar esse desafio, é essencial incentivar e apoiar a formação de mais profissionais qualificados em programação, através de metodologias atualizadas e mais direcionadas para as necessidades dessa área, de modo que seja possível desenvolver uma aprendizagem mais eficiente em Computação e áreas afins.

Diante desse contexto desafiador somado com a limitação de recursos e infraestrutura, é essencial buscar abordagens e estratégias que possam auxiliar tanto os alunos como os professores no processo de ensino-aprendizagem da Programação, para aprimorar a qualificação dos futuros desenvolvedores de software. Nesse sentido, a utilização de Objetos de Aprendizagem (OA) tem se mostrado uma alternativa promissora, como também o desenvolvimento de novos Objetos de Aprendizagem. Os OA são recursos educacionais digitais que visam apoiar o processo de ensino e aprendizagem por meio de elementos interativos, como animações, simulações e exercícios práticos (Souza *et al.*, 2020). Através dos OA, os alunos podem acessar conteúdos de forma mais visual, interativa e autônoma, o que pode contribuir para uma melhor compreensão e assimilação dos conceitos. Além disso, a implementação de metodologias de ensino inovadoras, que valorizam a participação ativa dos alunos e a resolução de problemas práticos, também são estratégias importantes para melhorar a qualidade do ensino da programação. (Rodrigues; Silva, 2019).

Neste Capítulo foram apresentados introdutoriamente os motivos que embasam o tema proposto, fornecendo suporte teórico para os objetivos que serão descritos nos subtópicos 1.1 e 1.2.

1.1. Objetivo Geral

Avaliar o impacto da utilização de Objetos de Aprendizagem no processo de ensino e compreensão da disciplina de Estrutura de Dados integrada ao curso de Ciência da Computação.

1.2. Objetivo Específicos

Para alcançar o objetivo geral definiu-se os seguintes Objetivos Específicos:

- Revisar o estado da arte sobre Objetos de Aprendizagem por meio de trabalhos relacionados à área da Ciência da Computação;
- Compreender as dificuldades enfrentadas pelos estudantes no processo de aprendizagem de Programação;
- Validar a utilização de Objetos de Aprendizagem como estratégia de ensino-aprendizagem na disciplina de Estrutura de Dados;
- Apresentar propostas de trabalhos futuros que ajude a desenvolver novas abordagens de ensino.

2. REFERENCIAL TEÓRICO

Neste Capítulo serão expostos os conceitos centrais que embasam esta pesquisa, explanando-se os conhecimentos e entendimento do tema de trabalho. O objetivo deste referencial teórico é oferecer uma visão panorâmica acerca da área da Programação e das principais perspectivas de ensino da Computação, buscando-se identificar as discussões mais importantes sobre os Objetos de Aprendizagem e sua aplicação no componente de Estruturas de Dados. Ao longo do capítulo serão apresentadas as seguintes subseções teóricas: Computação; Ensino da Programação; Objetos de Aprendizagem e Estruturas de Dados.

2.1. Educação e programação em Computação

A Computação desempenha um papel de extrema importância na sociedade contemporânea, impulsionando avanços tecnológicos que têm impacto direto na forma como se vive e se comunica (Campbell-Kelly *et al.*, 2014; Lyytinen *et al.*, 2016). Sua origem remonta ao século XX, aproximadamente entre as décadas de 1920 e 1930, e comparando a outras Ciências ela é muito recente, e surgiu como ideia de explorar máquinas capazes de executar operações complexas de forma automatizada.

Foi durante a Segunda Guerra Mundial que a Computação deu um salto significativo na História, ocorreram avanços com projetos secretos, como o Colossus, utilizados para decifrar códigos alemães (Copeland, 2010). Essas primeiras máquinas, conhecidas como computadores eletromecânicos, representaram uma aplicação prática da computação em larga escala.

Nos períodos subsequentes houve um rápido avanço, com desenvolvimento da Computação Eletrônica, com a criação dos primeiros computadores digitais programáveis, e o advento dos circuitos integrados na década de 1960, que possibilitou a criação de computadores menores, mais rápidos e mais eficientes (Campbell-Kelly, *et al.*, 2014). Neste ponto, a Computação se tornou acessível a empresas e instituições acadêmicas, impulsionando a pesquisa e o desenvolvimento de novas tecnologias e aplicações.

Na atualidade, a Computação engloba diversas áreas, incluindo Inteligência Artificial, Ciência de Dados, Segurança da Informação e Computação em Nuvem (Lyytinen, K. *et al.*, 2016). A rápida evolução tecnológica continua impulsionando a inovação em aplicações computacionais ressaltando ainda mais a importância da Computação na Sociedade Moderna.

No contexto do ensino da Computação, é frequente a abordagem de conceitos

fundamentais, tais como Algoritmos, Estruturas de Dados, Linguagens de Programação e Arquitetura de Computadores (Guimarães; Cardoso, 2017). Além disso, são exploradas metodologias e práticas relacionadas à resolução de problemas computacionais, desenvolvimento de software e Análise de Dados.

Esses recursos são essenciais para o entendimento e a capacitação dos alunos no curso de Computação. Uma abordagem fundamental é o Pensamento Computacional (PC). Essa abordagem consiste em um modo de pensar lógico e analítico, caracterizado pela representação de problemas em partes menores, identificação de padrões, abstração e criação de algoritmos para sua resolução (Ist; Csta, 2011).

O Pensamento Computacional desempenha um papel muito importante no desenvolvimento das habilidades de resolução de problemas, pensamento crítico e criatividade. Por meio dele, os alunos são capacitados para enfrentar desafios complexos em diversas áreas. Para Wing (2006) o Pensamento Computacional não é apenas valioso para Cientistas da Computação, mas é uma metodologia útil para diversos outros profissionais.

Em síntese, o ensino da Computação assume um papel fundamental na preparação de indivíduos para enfrentar os desafios tecnológicos do século XXI. Por meio do desenvolvimento de habilidades em Computação, os estudantes têm a oportunidade de se tornarem criadores de tecnologia, solucionadores de problemas e profissionais eficientes em um mundo cada vez mais digitalizado. Dessa forma, eles podem utilizar a Computação como uma ferramenta poderosa para apoiar a inovação, contribuir para a sociedade.

Uma vez que se analisa o ensino da Computação, é possível direcionar o foco para reflexões relevantes, considerando a importância da tecnologia na sociedade. No entanto, essa área enfrenta diversos desafios que exigem atenção e busca por soluções eficazes, a fim de garantir uma formação de qualidade para os alunos da área (Silva; Sousa, 2022).

Um dos desafios enfrentados no ensino da Programação é a complexidade dos conceitos e das estruturas envolvidas (Sentance; Csizmadia, 2017). O Aprendizado em Programação envolve a compreensão de Algoritmos, Lógica de Programação, linguagens específicas e resolução de problemas. Esses conceitos podem ser abstratos e desafiadores para os alunos, especialmente para aqueles que estão sendo apresentados pela primeira vez ao mundo da Programação.

A rápida evolução da tecnologia também representa um desafio para os educadores (Kafai; Burke, 2013). À medida que novas tecnologias surgem e as linguagens de programação são atualizadas, é necessário acompanhar essas mudanças para fornecer aos alunos conhecimentos atualizados e relevantes. Isso exige dos professores principalmente a adaptação

das metodologias de ensino.

Outro desafio é a disponibilidade de recursos adequados para o ensino da Programação (Hsu; Wu; Hwang, 2013). Nem todas as Instituições de ensino possuem infraestrutura e recursos tecnológicos suficientes para fornecer aos alunos um ambiente favorável para a aprendizagem da Programação. A falta de acesso a computadores, software e Internet de qualidade pode limitar as oportunidades de prática e dificultar a aplicação dos conhecimentos teóricos em projetos práticos.

Ademais, o ensino da Programação precisa também ser orientado a diversidade de perfis de estudantes (Grover; Pea, 2013). Os alunos têm diferentes estilos de aprendizagem, experiências prévias e motivações individuais. É fundamental adotar abordagens pedagógicas flexíveis e inclusivas, que atendam às necessidades e interesses variados dos alunos, os incentivando a participação, principalmente em projetos práticos. Para enfrentar esses desafios, é essencial investir em estratégias pedagógicas eficazes (Ragonis; Marques, 2019). Metodologias ativas, como aprendizagem baseada em projetos e resolução de problemas, podem proporcionar aos alunos uma experiência prática e contextualizada no aprendizado da Programação. O uso de recursos digitais interativos, tutoriais e ferramentas de apoio pode auxiliar no desenvolvimento das habilidades de Programação dos alunos.

De acordo com Gomes *et al.* (2008) e Santos e Costa (2006), o estudo da Programação transcende o domínio simples das Linguagens de Programação. Ele requer dos alunos uma capacidade sólida de abstração, resolução de problemas e raciocínio lógico, além de uma compreensão profunda e reflexiva dos conceitos, juntamente com práticas intensivas.

2.2. Estrutura de Dados

Estruturas de Dados (EDs) compreende maneiras de organizar e armazenar os dados/informações em uma unidade de armazenamento. Cormen (2002) ressalta que essa maneira de armazenar e organizar as informações tem como objetivo facilitar o acesso e a modificação destes dados. Para Backes (2017, p. 4) as estruturas de dados são “um relacionamento lógico entre diferentes tipos de dados visando a resolução de determinado problema de forma lógica”.

Existem vários tipos de EDs, cada uma logicamente definida e estruturada de formas diferentes umas das outras, possuindo custos de processamento e tempo de execução variados. Dentre as principais, destacam-se as estruturas lineares e as não lineares. Categoriza-se como lineares as pilhas, filas, listas e deque; e como não lineares as árvores, tabelas de hash e grafos.

É importante ressaltar que algumas estruturas possuem suas próprias variações como as listas que se dividem em encadeadas, duplamente encadeadas, circulares. Essas variações se distinguem na forma que os dados são inseridos, consultados e excluídos (Cormen *et al.*, 2009).

Estrutura de Dados é um dos componentes curriculares dos Cursos de Ciência da Computação, nele são apresentados os conceitos básicos de estruturas de dados, como *arrays* e listas, além de estruturas mais complexas como árvores e gráficos. Os alunos aprendem a armazenar e acessar elementos nessas estruturas, além de entender as diferenças entre elas (Goodrich *et al.*, 2013). Precisam compreender as características e propriedades dessas estruturas, bem como as operações que podem ser realizadas nelas (Sedgewick; Wayne, 2011).

Segundo Santos e Costa (2005), no processo de ensino e aprendizagem de Estrutura de Dados é preciso dar ênfase à parte conceitual e comportamental das EDs, para a partir daí seguir com a prática, codificar essas estruturas. Para a aplicação prática dos conceitos teóricos, os alunos têm a oportunidade de implementar as estruturas de dados aprendidas em Linguagens de Programação, como C++, Java ou Python. Isso permite que eles compreendam os detalhes de implementação e ganhem experiência.

2.2.1. Estrutura de Listas

A Estrutura de Dados desempenha um papel muito relevante na organização e manipulação de informações em programas de computador. Uma das estruturas de dados mais fundamentais é a lista dinâmica simplesmente encadeada. Neste subtópico, ser abordado o que são as listas dinâmicas simplesmente encadeadas, sua estrutura básica e as operações fundamentais associadas a elas.

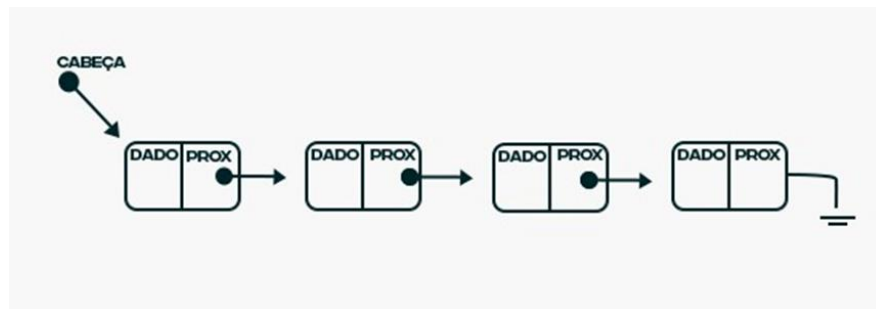
Uma lista dinâmica simplesmente encadeada é uma estrutura de dados linear que consiste em um conjunto de elementos chamados "nós", onde cada nó contém um valor de dados e um ponteiro (ou referência) para o próximo nó na lista (Cormen, *et al.* 2009).

Cada nó em uma lista dinâmica simplesmente encadeada possui duas partes principais: "Valor de Dados" que é onde os dados que você deseja armazenar na lista são mantidos. "Ponteiro para o Próximo Nó", este é um ponteiro (ou referência) que aponta para o próximo nó na lista. O último nó da lista aponta para um local nulo (*null*), indicando o fim da lista. Também conhecido como "Cabeça da Lista" que é o ponteiro responsável por guardar o endereço do primeiro nó da lista.

A estrutura de uma lista Simplesmente Encadeada visualmente representada na Figura 1

(Cormen *et al.*, 2009):

Figura 1 – Lista Simplesmente Encadeada



Fonte: Cormen *et al.*, 2009:

Algumas das operações fundamentais que podem ser executadas em listas dinâmicas simplesmente encadeadas: Inserção (adiciona um novo nó à lista, seja no início, no final ou em uma posição específica); Exclusão (remove um nó da lista, geralmente com base em seu valor de dados ou posição); Busca (procurar um valor de dados específico na lista); Travessia (percorrer a lista, geralmente para imprimir seus elementos ou realizar alguma operação em cada nó); Atualização (modifica o valor de dados de um nó existente). Essas operações são essenciais para a manipulação eficaz de dados em listas dinâmicas simplesmente encadeadas e têm um amplo espectro de aplicações em Programação.

O Aprendizado em Estrutura de Dados é essencial para qualquer estudante de Ciência da Computação. Ele fornece as bases para a organização eficiente de informações e é fundamental na resolução de problemas computacionais complexos. É um componente essencial na formação de profissionais de informática.

2.3. Objetos de Aprendizagem

Conforme a definição do *Learning Object Metadata Working Group* (IEEE, 2005), os Objetos de Aprendizagem (OA) são definidos como "entidades digitais ou não digitais que podem ser utilizadas, reutilizadas ou referenciadas durante a aplicação de tecnologias de suporte ao ensino".

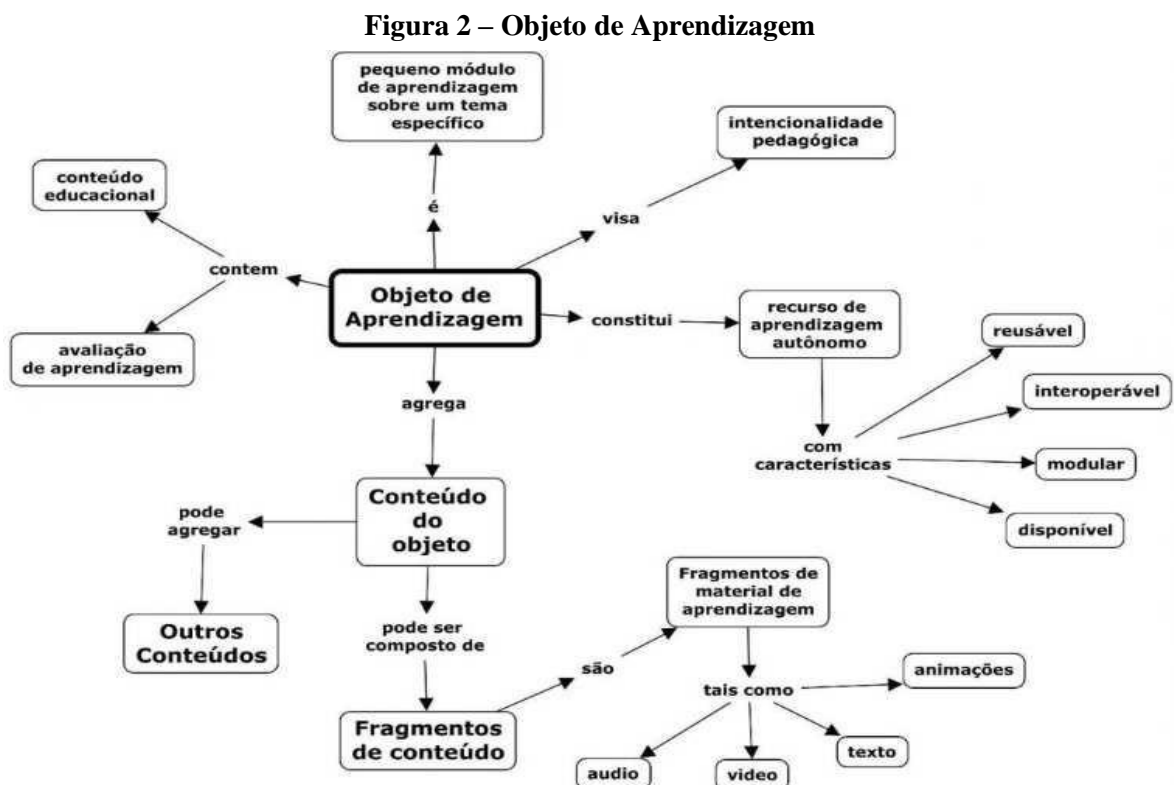
Os OA surgiram com a necessidade de tornar os métodos tradicionais de ensino mais interativos e atrativos, com o intuito de apoiar e auxiliar pessoas que necessitam de métodos alternativos de ensino. Tiosso *et al.* (2014) ressaltam que os OA não substituem os professores, mas atuam como um apoio na aprendizagem corroborando para a participação mais efetiva dos discentes nas disciplinas de ensino.

O uso generalizado de Objetos de Aprendizagem é acompanhado por uma ampla

disponibilidade de repositórios que abrigam diversos tipos de OA (Tarouco, *et al.*,2021). Além disso, foram aprimoradas ferramentas para auxiliar não apenas na criação desses objetos, mas também em sua catalogação em repositórios, a fim de facilitar a busca e o reuso.

Devido às expectativas e demandas dos alunos em relação ao conteúdo e à forma de exploração desses objetos, encontrar o objeto de aprendizagem desejado para uma determinada situação de aprendizagem e contexto é um processo desafiador. No Mapa Conceitual a seguir (Figura 2), observa-se como a essência da ideia relacionada aos Objetos de Aprendizagem pode ser resumida:

A Figura 2 fornece um mapa conceitual que esclarece as principais características de um Objeto de Aprendizagem. Este é definido como um pequeno módulo de ensino que se concentra em um tema específico, evidenciando sua intencionalidade pedagógica. O OA contém não apenas informações educacionais, mas também elementos de avaliação de aprendizagem, permitindo aos alunos testar e consolidar seu conhecimento. Além disso, a figura destaca a capacidade de um objeto de aprendizagem de agregar outros conteúdos ou fragmentos de conteúdo, tornando-o flexível e adaptável a diversas necessidades educacionais. A figura destaca que um objeto de aprendizagem constitui um recurso de aprendizagem independente, o que significa que ele pode ser utilizado de forma independente pelos alunos, facilitando a aprendizagem autodirigida e a exploração do conhecimento.



Fonte: Tarouco, *et al.* (2021)

São muitos os tipos de Objetos de Aprendizagem, porém os mais utilizados no ensino da Computação que podem ser destacados são os jogo/game, ferramentas de desenvolvimento, ferramentas de avaliação, simulação, texto/artigo, curso online, ferramentas para prática autônoma, tarefa/atividade, e representação de contexto. Dentre os métodos mencionados, destaca-se o uso de jogos como uma estratégia popular para captar a atenção e despertar o interesse dos alunos, o que resulta em uma curva de aprendizado mais suave. Essa abordagem é corroborada por Souza *et al.* (2013), que enfatiza o potencial dos objetos de aprendizagem baseados em jogos para facilitar o processo de ensino e aprendizagem.

2.4. Computação Desplugada

A metodologia de Computação Desplugada é uma abordagem que se concentra no ensino de conceitos de Ciência da Computação e Programação sem a necessidade de acesso constante à Internet ou dispositivos online (Kafai; Burke, 2015). Essa metodologia é particularmente relevante em contextos nos quais a conectividade é limitada ou inexistente, mas também é valiosa como complemento ao ensino de informática em ambientes tradicionais.

A base dessa metodologia está na criação de atividades práticas e interativas que utilizam recursos físicos, como papel, lápis, jogos de tabuleiro e outros materiais tangíveis, para ensinar princípios fundamentais da Ciência da Computação. Essas atividades são projetadas para serem acessíveis a todos, independentemente do acesso à tecnologia digital.

Na Computação Desplugada, usam-se quebra-cabeças lógicos e jogos de cartas para ensinar Algoritmos, Estruturas de Dados e resolução de problemas. Além disso, os alunos desenvolvem algoritmos e depuram códigos manualmente como parte essencial dessa metodologia.

A abordagem de Computação Desplugada oferece uma série de benefícios, incluindo a promoção do pensamento lógico e da resolução de problemas, o desenvolvimento da criatividade e a familiarização com os conceitos fundamentais da Ciência da Computação. Isso a torna uma ferramenta valiosa tanto para a educação formal, em sala de aula, quanto para iniciativas de ensino de Programação em ambientes informais.

A metodologia de Computação Desplugada oferece uma solução inclusiva e versátil, permitindo que pessoas de todas as idades e níveis de experiência adquiram conhecimentos sólidos nesta área da Computação, independentemente da disponibilidade de recursos digitais.

3. TRABALHOS RELACIONADOS

Neste Capítulo serão apresentados os principais estudos relacionados ao tema da pesquisa, os quais foram obtidos e revisados por meio de uma Revisão Sistemática da Literatura.

Tornou-se evidente que com o passar do tempo foi detectado a necessidade de desenvolver e aplicar novos métodos de ensino para diferentes áreas, principalmente aquelas relacionadas com o ensino da Computação. Entre os trabalhos que mencionam a aplicação de jogos como ferramenta de ensino, destacam-se Wong e Yatim (2013) e Rodríguez, *et al.* (2020).

Referente à análise de metodologias ativas aplicadas ao ensino, é possível evidenciar o trabalho de Ribeiro e Passos (2020). Pelo levantamento da eficácia educacional de Objetos de Aprendizagem, destaca-se o trabalho de Antônio (2018). E a respeito da criação de novos métodos de ensino no apoio ao ensino da Estrutura de Dados, destaca-se a pesquisa de Sousa (2019). É importante ressaltar que a seleção deste último ocorreu de forma paralela aos demais artigos já citados, tendo em vista o que foi lido e revisado antes do planejamento da revisão sistemática.

Seng e Yatim (2013), apresentam o jogo “Ztech” como ferramenta de ensino para a aprendizagem para orientação a objetos. O jogo foi desenvolvido para servir de ferramenta de ensino, sendo composto por cinco níveis direcionados para o aprendizado dos fundamentos acerca de Orientação a Objetos, como encapsulamento, herança, polimorfismo e abstração. O jogo também conta com alguns conceitos básicos de programação. O estudo contou com a participação de 40 alunos de dois programas (Bacharel em Ciência da Computação e Bacharel em Engenharia de Software) e abordou o módulo denominado Introdução à Programação Orientada a Objetos. De acordo com os dados coletados, 86% dos alunos apontaram que a abordagem de aprendizagem baseada em jogos é melhor do que a abordagem tradicional e concluem que jogos de computador podem ser ferramentas úteis à aprendizagem.

A pesquisa de Rodríguez, *et al.* (2020) traz uma análise na experiência de aprendizagem baseada em jogos na *web* utilizando um jogo chamado “eLiza”. O eLiza é uma ferramenta educacional semelhante ao *Kahoot* no qual o professor pode alimentar o programa com questões, determinar tempo e peso para cada questão. Ao final do estudo foi apontado que mais de 75% dos alunos consideraram que a utilização do eLiza foi positiva para melhorar o processo de ensino e aprendizagem.

Ribeiro e Passos (2020) apresentaram um estudo sobre as metodologias ativas aplicadas

ao processo de ensino e aprendizagem na área da Informática. O estudo levantou os tipos mais utilizados de Metodologias Ativas que estão sendo utilizadas no ensino e aprendizagem em Ciência da Computação, bem como suas desvantagens e seus benefícios. As autoras concluíram que as metodologias ativas possuem um impacto positivo no processo de ensino e aprendizagem em Ciência da Computação.

Kalloniatis (2018) buscou avaliar a eficácia educacional e o apelo motivacional dos Objetos de Aprendizagem para o ensino de conceitos de programação de computadores em salas de aula do ensino médio. O autor afirma que o uso de objetos de aprendizagem tem sido experimentado como uma importante ferramenta de aprendizagem da educação em Ciência da Computação e passou a desempenhar um papel cada vez mais central na aprendizagem em sala de aula. Os resultados indicaram que os alunos que usaram os objetos de aprendizagem tiveram um desempenho significativamente melhor do que aqueles que não usaram, além disso, eles relataram maior satisfação com as atividades e melhor interesse em continuar estudando programação. No entanto, o autor ainda ressalta que as evidências empíricas sobre a eficácia educacional dos OA em turmas de ensino médio são limitadas e necessita de mais experimentações.

Sousa (2019), apresenta em seu trabalho uma estrutura para ensinar e aprender conceitos básicos de Estrutura de Dados, aplicando recursos lúdicos para motivar a participação dos alunos. Após uma análise horizontal sobre o cenário de aprendizagem da computação, mais especificamente da disciplina de Estrutura de Dados, Sousa aponta que os altos índices de evasão dos alunos nos períodos iniciais de cursos relacionados à Computação decorrem da dificuldade com a abstração, raciocínio lógico e desenvolvimento de algoritmos, e entende que é necessário a aplicação de novas metodologias para apoiar os professores na tarefa de transmitir o conhecimento de maneira mais eficiente.

Um dos destaques do trabalho foi a didática aplicada para apoiar o ensino de estrutura de dados do tipo lista, utilizando os conceitos de computação desplugada utilizando as carteiras da sala de aula para explicar e mostrar como as operações que envolvem a representação de uma lista.

A partir dos trabalhos apresentados neste Capítulo, fica evidente a necessidade de adotar novas metodologias de ensino. Esses estudos demonstram a existência de diversas abordagens e pesquisas que exploram o uso de jogos como ferramentas de ensino, a análise de metodologias ativas aplicadas ao processo de ensino e aprendizagem e a avaliação da eficácia educacional de Objetos de Aprendizagem. Essas descobertas ressaltam a importância de se buscar alternativas inovadoras e eficazes para aprimorar a aprendizagem na área da Computação.

O trabalho realizado por Sousa (2019) desempenha um papel de extrema importância na pesquisa, uma vez que apresenta uma metodologia inovadora baseada em objetos de aprendizagem e computação desplugada para aprimorar o ensino na disciplina de Estrutura de Dados. A relevância deste estudo reside no fato de que a metodologia proposta ainda não foi testada neste contexto específico. Isso identifica uma lacuna significativa no conhecimento, considerando a importância da disciplina de Estrutura de Dados na ciência da computação. A introdução de abordagens pedagógicas inovadoras pode ter um impacto substancial na qualidade do aprendizado dos alunos.

Neste Capítulo foram apresentados e discutidos trabalhos relacionados ao tema proposto, fornecendo embasamento teórico para os procedimentos de realização da pesquisa. Essa revisão foi fundamental para o planejamento da Metodologia que será apresentada no próximo capítulo.

4. METODOLOGIA

Neste Capítulo serão apresentados os métodos utilizados para coletar os dados da pesquisa e tem como objetivo descrever detalhadamente os procedimentos adotados para analisar e interpretar os dados essenciais para responder às Questões de Pesquisa e atingir os objetivos definidos no Trabalho. A seção se divide nos seguintes tópicos: *Design Science*, Revisão Sistemática de Literatura e Design do Experimento.

O *Design Science* (DS), também conhecido como Ciência do *Design*, é uma abordagem metodológica que combina princípios e práticas de pesquisa científica com o processo de desenvolvimento de soluções práticas e inovadoras para problemas complexos. O *Design Science* envolve uma sequência cíclica de atividades, incluindo a identificação e compreensão do problema, a proposição de uma solução em forma de válvula de escape, a construção da válvula de escape e sua avaliação em relação a critérios de desempenho e eficácia (Hevner *et al.*, 2004).

Essa abordagem se baseia em princípios científicos, mas também incorpora a criatividade e o conhecimento prático para desenvolver soluções viáveis e inovadoras. O objetivo do *Design Science* é criar soluções práticas que possam ser integradas e utilizadas no mundo real, garantidas para a resolução de problemas e melhoria de processos e sistemas. Além disso, a abordagem valoriza a comunicação e a disseminação dos resultados, permitindo que outros pesquisadores e profissionais possam se beneficiar das soluções avançadas.

O livro "Design Science Research: Método de Pesquisa para o Avanço da Ciência e Tecnologia", de autoria de Aline Dresch, Daniel Pacheco Lacerda e José Antonio Valle Júnior Antunes, explora o conceito de pesquisa em DS, fornecendo orientações sólidas para a condução de estudos interdisciplinares e inovadores. Escrito por autores experientes na área, a obra destaca a importância da pesquisa baseada em design e sua significativa contribuição para o avanço do conhecimento e a promoção da inovação.

4.1. Revisão Sistemática de Literatura

Para fins de revisão bibliográfica foi utilizada a metodologia de pesquisa denominada Revisão Sistemática de Literatura (RSL). Uma Revisão Sistemática de Literatura “é um meio de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes para uma determinada questão de pesquisa, ou tópicos de área, ou fenômeno de interesse”.

A RSL ou de modo simplificado Revisão Sistemática é um estudo secundário que faz uso de uma metodologia bem definida para coletar, analisar e interpretar os dados de estudos

primários relevantes para um determinado ponto de pesquisa. É importante pontuar que uma RSL é definida em três fases principais, denominadas de Planejamento, Condução e Relatório. A etapa de Planejamento tem como objetivo especificar os procedimentos necessários para a revisão sistemática. Na etapa de Desenvolvimento da RSL é efetuada triagem dos artigos de acordo com os critérios definidos na etapa de Planejamento. Na última etapa, é executada a retirada dos dados mais relevantes, direcionados a responder às Questões da Pesquisa.

4.1.1. Planejamento da RSL

É na fase de Planejamento que são definidos os objetivos da Revisão Sistemática, bem como as *strings* de busca, os bancos de dados (BD) e a fixação dos critérios de inclusão (CI) e/ou exclusão (CE). A principal etapa no planejamento compreende a definição do protocolo da RLS. Para Kitchenham (2007) ao fazer isso diminuimos a incidência de resultados tendenciosos.

A presente pesquisa tem como objetivo verificar a eficácia da aplicação de Objetos de Aprendizagem (OA) no ensino da Computação, com base nos procedimentos metodológicos de Sousa (2019), além de investigar e entender o impacto dos OA no Ensino de Estrutura de Dados. Este trabalho se propõe a responder as seguinte Questões de Pesquisa:

Questão Geral de Pesquisa (QGP): Como aplicar e avaliar o uso de Objetos de Aprendizagem no processo de ensino/aprendizagem de Estrutura de Dados no Curso de Ciência da Computação?

Decorrente da QGP foram definidas as seguintes Questões Secundárias de Pesquisa (QSP):

QSP1: Quais são os principais tipos de Objetos de Aprendizagem que podem ser utilizados no apoio ao ensino da Computação?

QSP2: Como avaliar o impacto dos OA no Ensino da Computação?

QSP3: Quais procedimentos de ensino da Programação podem ser transformados com a aplicação do OA?

A discussão a respeito de cada questão de pesquisa será apresentada no Capítulo de Resultados desse trabalho.

4.1.2. Desenvolvimento da RSL

Nesta etapa da RSL foi realizada a seleção dos trabalhos a serem revisados, utilizando os repositórios de publicações, com aplicação de *string* de busca e de critérios de inclusão e

exclusão. É importante ressaltar que nas bases de pesquisas já citadas foram filtrados apenas trabalhos dos anos de 2013 a 2023. Para prospecção dos artigos foram utilizados os seguintes repositórios: *ScienceDirect*, *Google Scholar* e *ACM Digital Library*.

Com o objetivo de filtrar os trabalhos relacionados a esta pesquisa, foi aplicada a seguinte *string* de busca ["Ensino da Programação" OR "*Teaching Programming*") AND ("Aprendizagem Efetiva" OR "*Effective Learning*") AND ("Objetos de Aprendizagem" OR "*Learning Objects*") AND ("Estrutura de Dados" OR "*Data Structures*") AND ("Listas Simplesmente Encadeadas" OR "*Singly Linked Lists*")].

Após a busca dos trabalhos nas respectivas Bases, foi iniciado processo de revisão concentrando-se na leitura do título e do resumo das publicações, elegendo-se os trabalhos que iriam para a próxima etapa. Posteriormente, foi desenvolvida a leitura da Introdução e da Conclusão, aplicando uma filtragem dos trabalhos obtidos. A revisão bibliográfica dos trabalhos retornados em cada base foi refinada a partir da aplicação dos seguintes Critérios de Exclusão (CE):

- **CE1** - Trabalhos que não abordam o ensino da Computação, Estrutura de Dados ou Objetos de Aprendizagem.
- **CE2** - Trabalhos que são iguais ou que possuam versão mais atualizada (deve ser mantido o mais recente).
- **CE3** - Trabalhos que são um resumo ou uma Revisão Sistemática de Literatura.
- **CE4** - Trabalhos escritos em outros idiomas além do Inglês ou Português.
- **CE5** - Trabalhos que não estão disponíveis para visualização ou download no momento da pesquisa.
- **CE6** - Trabalhos incompletos.

4.2. Design do Experimento

Nesta seção, serão delineadas as fases envolvidas no experimento, abrangendo desde a etapa de planejamento, a descrição do Objeto de Aprendizagem designado para a aplicação, o processo de seleção dos participantes e, por condução da fase executiva do experimento.

O foco do experimento foi direcionado para a avaliação teórica e prática no contexto da disciplina de Estrutura de Dados. Nossa pesquisa visa aprofundar a compreensão dos impactos da aplicação de um Objeto de Aprendizagem específico, direcionado à exploração das Listas Simplesmente Encadeadas. Esta forma de estrutura de dados é de suma importância no desenvolvimento de Algoritmos e Sistemas, e o experimento tem o objetivo de avaliar como o

uso de um Objeto de Aprendizagem pode influenciar a compreensão, o aprofundamento do conhecimento e o desempenho dos estudantes no âmbito do estudo sobre Listas Dinâmicas.

4.2.1. Planejamento do Experimento

Antes da elaboração da estrutura do experimento, a etapa inicial compreendeu a seleção da disciplina em que o Objeto de Aprendizagem seria implementado. No contexto desta pesquisa, foi determinado que a disciplina de *Estrutura de Dados* (ED) seria incluída, devido à sua carga teórica e prática específica, na qual os discentes enfrentam frequentemente desafios na transposição dos conceitos teóricos para a aplicação prática da Programação.

Após escolha da disciplina alvo do experimento, uma etapa subsequente envolveu a escolha do Objeto de Aprendizagem que seria incorporado ao currículo da disciplina de Estrutura de Dados, bem como a determinação do enfoque da aplicação deste objeto designado. Para a pesquisa em questão, optou-se por verificar a aplicação do Objeto de Aprendizagem (OA) desenvolvido no artigo intitulado 'Computação Desplugada no Ensino de Estrutura de Dados', de autoria de Sousa (2019). O referido OA se caracteriza por abordar a aplicação de uma dinâmica que faz uso do ambiente da sala de aula e envolve os alunos no processo de ensino, com o propósito de transmitir e representar os conceitos, a estrutura e as operações fundamentais relacionadas às Listas Simplesmente Encadeadas.

As etapas finais do planejamento do experimento englobam a seleção das turmas participantes e a definição da execução do experimento, que ocorreu nos dias 16 e 17 de outubro de 2023.

4.2.2. Estrutura do Objeto de Aprendizagem

O Objeto de Aprendizagem (OA) selecionado é fundamentado na metodologia de Computação Desplugada. O OA tem a finalidade de facilitar a compreensão dos princípios fundamentais das Listas Simplesmente Encadeadas e das operações realizadas em tempo de execução.

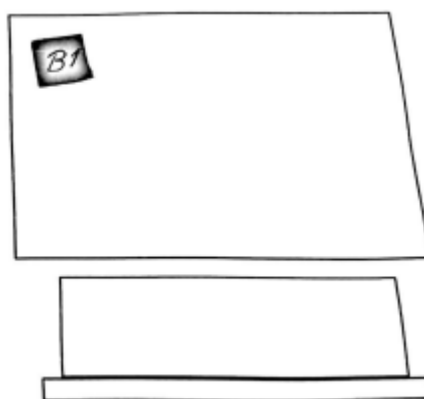
O cenário de aprendizagem se define da sala de aula e da participação dos alunos, com o propósito de criar uma dinâmica interativa que contribua para a consolidação desses conceitos. Tal abordagem visa apoiar tanto o docente quanto o discente no seu processo de ensino e aprendizagem respectivamente.

Este OA é construído em torno de uma estrutura relativamente simples, na qual os

estudantes desempenham papéis fundamentais. Neste contexto, os alunos representam os nós da lista e os ponteiros auxiliares, enquanto um deles é designado como o "sistema" que controla as operações. Além disso, as carteiras na sala de aula são utilizadas para representar os espaços de memória de um computador, e para isso elas devem estar devidamente etiquetadas conforme ilustrado na Figura 3.

A Figura 3 mostra uma representação da maneira como as cadeiras devem ser identificadas na sala de aula, utilizando os papéis adesivos, comumente conhecidos como post-its.

Figura 3 - Representação da identificação das cadeiras

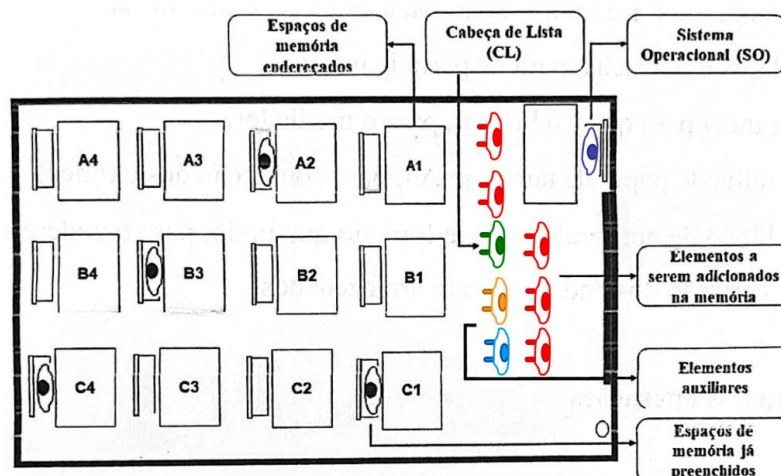


Fonte: Sousa (2019)

Antes de iniciar a dinâmica é necessário preparar o cenário. A Figura 4 mostra como a sala deve ser organizada: cadeiras alinhadas e identificadas; os estudantes que irão representar os elementos da Lista a serem adicionados na memória, em pé na frente dos demais, que continuam sentados (representando posições de memória ocupadas no computador).

Para que a dinâmica funcione de maneira eficiente, a participação de pelo menos nove alunos é necessária. Um deles representa o Sistema, outro atua como a cabeça da Lista e dois desempenham o papel dos ponteiros auxiliares. A Figura 4 apresenta o cenário simplificado da dinâmica.

Figura 4 – Descrição do cenário



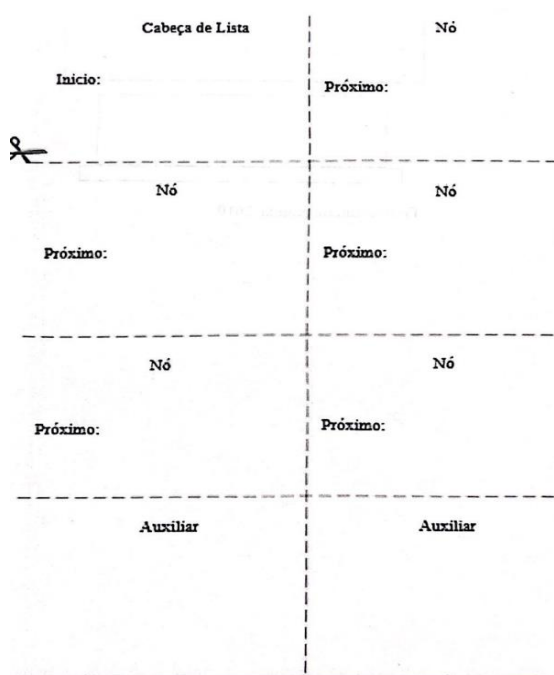
Fonte: Sousa (2019)

Os cinco alunos restantes representam os nós que serão manipulados de acordo com as instruções do Sistema. Todos esses alunos devem estar equipados com lápis, borracha e o cartão descrito na Figura 5. O cartão desempenha um papel essencial, permitindo que os alunos estabeleçam conexões entre os nós e naveguem a partir da cabeça da lista até os demais nós, seguindo as informações presentes no cartão de cada nó inserido na lista.

É importante ressaltar que a configuração do cenário pode ser ajustada de acordo com o número de alunos interessados em participar, o tempo disponível para a dinâmica e a abordagem lógica que os alunos desejam empregar para realizar as operações. Esta adaptabilidade visa proporcionar uma experiência mais personalizada e alinhada às necessidades específicas do grupo, assegurando a eficácia e a participação ativa dos alunos no contexto da atividade proposta.

A Figura 5 exibe o formato e as dimensões dos cartões utilizados pelos participantes durante a dinâmica. Este formato deve ser utilizado para uma execução com 8 participantes, caso haja um número maior de participantes, deve-se utilizar uma folha adicional somente com o modelo referente aos nós.

Figura 5 - Modelo dos cartões em folha A4



Fonte: Sousa (2019)

Essa abordagem interativa visa tornar o processo de aprendizagem das Listas Simplesmente Encadeadas mais envolvente e prático, possibilitando que os alunos visualizem, compreendam e participem ativamente na construção e manipulação dessas estruturas de dados.

4.2.3. Participantes do Experimento

Com o objetivo de conduzir o experimento, optou-se por selecionar duas turmas do Curso de Ciência da Computação na Universidade Estadual da Paraíba (Campus VII), uma no período matutino, denominada Turma A, e outra no período noturno, denominada Turma B. O experimento foi realizado com uma amostra definida por 64 alunos, divididos igualmente, com 32 estudantes frequentando o período matutino e 32 no período noturno. Isso permitiu que fosse possível analisar a relevância do OA escolhido na etapa de planejamento, aplicando-o em uma das turmas e mantendo a outra como grupo de controle, possibilitando uma análise comparativa dos dados.

Todos os discentes envolvidos no estudo são, invariavelmente, estudantes matriculados na disciplina de Estrutura de Dados, independentemente do período em que estejam cursando a referida disciplina.

4.2.4. Execução do Experimento

A execução do experimento foi dividida em três partes, cada uma delas com procedimentos específicos e um material de trabalho elaborado fisicamente. Na primeira parte, iniciamos com a aplicação de uma aula expositiva abordando os assuntos referente ao plano de aula conforme descrito no Apêndice A, seguida pela realização da dinâmica planejada (OA). A dinâmica orientada por Sousa (2019) teve o propósito de enriquecer a compreensão do conteúdo discutido na sala de aula. Após a conclusão da dinâmica, procedemos à administração de um questionário, conforme estabelecido no Apêndice C. Esse questionário estava diretamente relacionado ao assunto abordado em sala de aula e destinava-se a avaliar o nível de absorção do conteúdo pelos alunos. Suas questões foram formuladas de modo a abranger os tópicos discutidos durante a aula, permitindo uma avaliação precisa do grau de entendimento e retenção por parte dos estudantes.

A segunda parte do experimento, que foi realizado com a Turma B, por sua vez, consistiu exclusivamente na execução do plano de aula, conforme documentado no Apêndice B, seguida pela aplicação do mesmo questionário relacionado ao assunto abordado em sala de aula. É importante ressaltar que, ao contrário da primeira parte, esta etapa não incluiu a realização da dinâmica com a turma. A intenção era avaliar especificamente o impacto do plano de aula no processo de absorção do conteúdo pelos alunos.

Além disso, a decisão de conduzir a primeira parte do experimento com a Turma A, pertencente ao período matutino, e a segunda parte com a Turma B, do período noturno, teve como motivação a consideração de fatores externos, como as limitações de tempo disponível para cada turma. Essa abordagem permitiu uma análise comparativa dos resultados, levando em consideração possíveis variações nos níveis de atenção e absorção do conteúdo devido ao horário das aulas, bem como a eficácia do plano de aula em diferentes contextos temporais.

Na terceira fase do experimento, foi realizada a correção dos questionários que foram distribuídos nas turmas. Durante essa etapa, foi realizada uma meticulosa catalogação do número de respostas corretas para cada questão, segmentando esses dados por turma.

Para a etapa de correção do questionário, as perguntas discursivas foram avaliadas com base na precisão abrangente das respostas dos alunos, classificando as respostas como corretas somente quando o estudante apresentou uma compreensão integral e precisa da questão. Qualquer desvio ou erro em qualquer parte da resposta resultou na consideração da questão como incorreta. Esse procedimento permitiu uma análise comparativa do desempenho dos estudantes em relação às diferentes questões e turmas, fornecendo informações essenciais para

a avaliação dos resultados do estudo.

Nesta seção, delineamos as várias etapas envolvidas no design e planejamento do experimento, desde a fase inicial de concepção até a implementação prática. No próximo Capítulo, o Capítulo de Resultados, serão apresentados os resultados obtidos da aplicação do experimento.

5. RESULTADOS

Neste Capítulo serão apresentados e discutidos os resultados da Pesquisa, revelando as descobertas que surgiram a partir da análise dos dados coletados com o experimento do Objeto de Aprendizagem. Estes resultados fornecerão respostas diretas às Questões de Pesquisa que nortearam o estudo, destacando tendências e padrões identificados ao longo da pesquisa. Além disso, serão abordadas as implicações e a relevância dessas descobertas para o campo de estudo, contribuindo para o avanço do conhecimento acadêmico e sua aplicação prática.

5.1. Resultado da RSL

Na Tabela 1 é apresentado o espelho da fase de condução da pesquisa, contendo a quantidade de artigos retornados na *string* de busca em cada Base de dados que foi feita a filtragem, a quantidade de artigos rejeitados em cada Critério de Exclusão, por fim a totalização dos trabalhos incluídos.

Tabela 1. Triagem da RSL

| Bases | Aplicação da String | CE1 | CE2 | CE3 | CE4 | CE5 | CE6 | Incluídos |
|------------------------------------|--------------------------------|------------|------------|------------|------------|------------|------------|------------------|
| <i>Science Direct</i> | 7 | 3 | 0 | 0 | 0 | 1 | 0 | 3 |
| <i>Google Scholar</i> | 130 | 43 | 7 | 43 | 5 | 10 | 9 | 13 |
| <i>Acm Digital Library</i> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Total | 137 | 46 | 7 | 43 | 5 | 11 | 9 | 16 |

De acordo com a Tabela 1, utilizando a *string* de busca definida, foram encontrados inicialmente 137 artigos. Após a aplicação dos Critérios de Exclusão (CE), foram selecionados 16 trabalhos para uma leitura completa. Dentre esses 16 artigos, foram identificados quatro trabalhos que apresentam um alinhamento metodológico mais relacionado com os objetivos

com deste trabalho.

5.2. Questões da Pesquisa

No subtópico 4.1, intitulado "Revisão Sistemática de Literatura", foram formuladas, durante a fase de Planejamento, questões que direcionaram a elaboração da RSL. Este segmento abordará as respostas para essas indagações, as quais foram obtidas por meio da execução da Revisão Sistemática de Literatura.

QSP1: Quais são os principais tipos de Objetos de Aprendizagem que podem ser utilizados no apoio ao ensino da Computação?

Ao examinar as obras, tornou-se evidente que os Objetos de Aprendizagem abrangem diversos tipos, dentre os quais os fundamentados em abordagens de jogos demonstraram resultados altamente positivos quando trabalharam no ensino da Computação, conforme evidenciado nas pesquisas realizadas por Wong e Yatim (2013), Seng e Yatim (2013), e Rodríguez, Antón et al. (2020).

QSP2: Como avaliar o impacto dos OA no Ensino da Computação?

A validação do impacto dos Objetos de Aprendizagem (OA) é predominantemente realizada através da avaliação do envolvimento do aluno, seu nível de satisfação e desempenho. Este processo é frequentemente demorado por meio da aplicação de questionários estruturados ou avaliações específicas. Conforme evidenciado nas Pesquisas de Seng e Yatim (2013), Rodríguez, Antón et al. (2020), Kalloniatis (2018).

QSP3: Quais procedimentos de ensino da Programação podem ser transformados com a aplicação do OA?

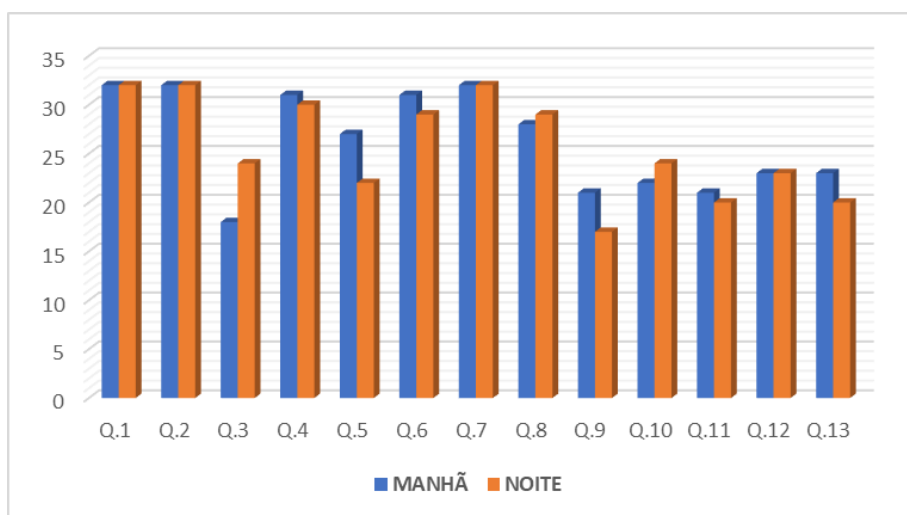
Os métodos tradicionais de ensino possuem suas limitações e a utilização de Objetos de aprendizagem como meio complementar ao método tradicional vem se mostrando promissor. Métodos complementares que não sejam somente teorias, práticas e avaliações, mas abordagens mais integrativas que despertem o interesse e a atenção do aluno. Conforme evidenciado nas Pesquisas de Ribeiro e Passos (2020), Rodríguez, Pérez-Juárez, Díaz-Pernas, González-Ortega, Martínez-Zarzuela e Aguiar-Pérez (2020), Seng e Yatim (2013), Kalloniatis (2018) e Sousa (2019)

5.1. Resultados do Experimento

Os resultados do experimento revelam uma discreta, porém perceptível diferenciação entre a turma matutina (Turma A) e a turma noturna (Turma B), conforme evidenciado no gráfico representado no Gráfico 1. A turma matutina, embora não tenha demonstrado uma diferença expressiva, obteve um desempenho superior em termos de acertos quando comparada à turma noturna, que, é relevante salientar, participou exclusivamente da aula expositiva.

O Gráfico 1 fornece uma representação visual da comparação da margem de acertos das questões entre as turmas da manhã e da noite.

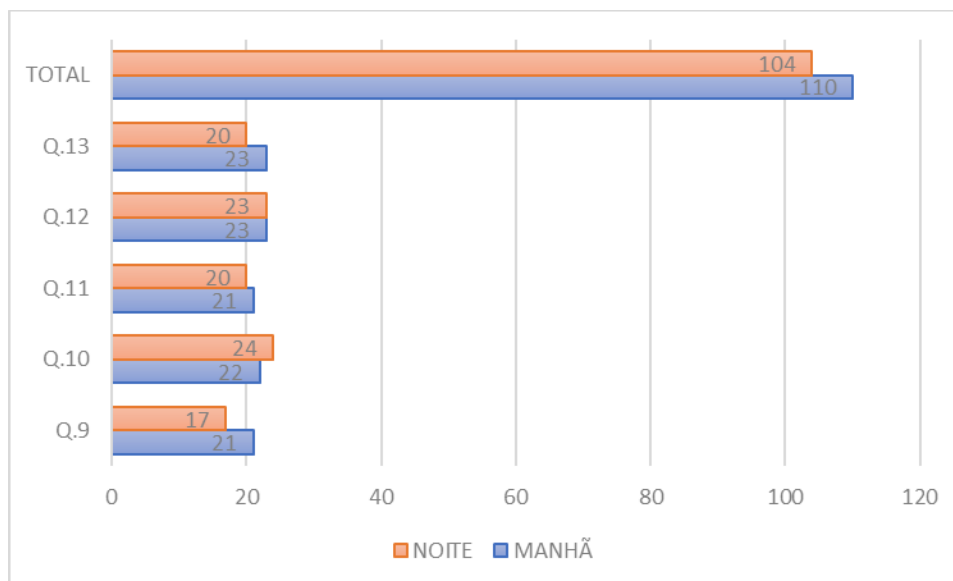
Gráfico 1 – Quantidade de acertos de cada questão por turma



Fonte: Autoria própria, 2023

Destaca-se, sobretudo, a análise das questões abertas apresentadas no questionário, correspondendo a 38% da avaliação e identificada pelas questões numeradas de nove a treze. Neste aspecto, observa-se um desempenho superior da Turma A em comparação com a Turma B, conforme evidenciado no Gráfico 2.

A Gráfico 2 apresenta um gráfico que fornece uma representação visual da comparação da margem de acertos das questões abertas (Ver Apêndice B) entre as turmas da manhã e da noite.

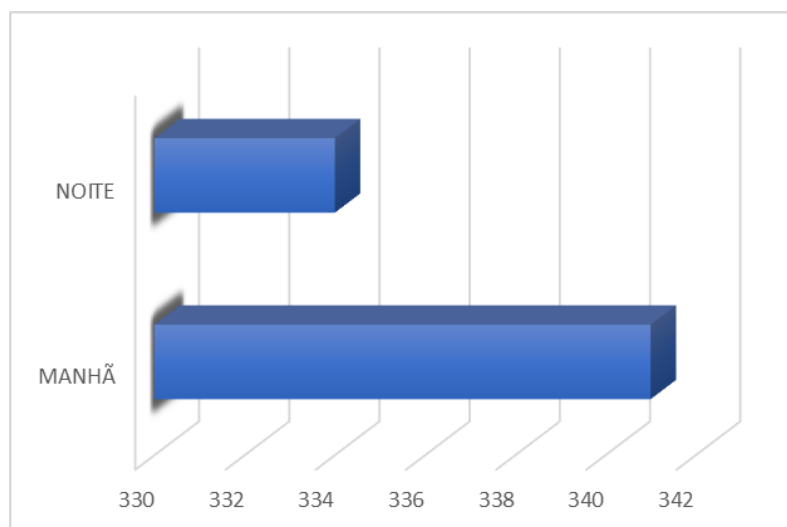
Gráfico 2 – Quantidade de acertos das questões abertas por turma

Fonte: Autoria própria, 2023

No Gráfico 2, é visível que a Turma A demonstrou uma diferença compreensível em relação à Turma B, no que diz respeito à quantidade de acertos nas questões abertas. Os resultados apontam para um melhor desempenho da Turma A nesse tipo específico de questão, demonstrando um nível mais elevado de proficiência na compreensão e aplicação do conteúdo ensinado em comparação com a Turma B. Suas respostas ressaltam uma precisão e detalhamento superiores em relação à Turma B. Isso sugere que a abordagem pedagógica empregada, que incluiu o plano de aula e a dinâmica, aplicada à Turma A, desempenhou um papel significativo na absorção eficaz do conteúdo de Listas.

É relevante ressaltar que as questões abertas demandam maior perspicácia e domínio do conteúdo lecionado durante a aula. Nesse contexto, os alunos foram desafiados a comunicar, com suas próprias palavras, os procedimentos e abordagens que adotariam para atingir os objetivos propostos pelas questões. Essa modalidade de questionamento não oferece apenas uma compreensão profunda do assunto, mas também a capacidade de aplicar o conhecimento de maneira prática e criativa, demonstrando um nível mais avançado de aprendizado.

Ao analisar o total de acertos de cada turma, também é evidente que a Turma A obteve um número superior de respostas corretas em comparação com a Turma B, conforme ilustrado no Gráfico 3.

Gráfico 3 – Quantidade total de acertos por Turma

Fonte: Aatoria própria, 2023

Adicionalmente, ao longo do procedimento experimental, foi observado que a Turma A demonstrou uma participação mais ativa, especialmente durante a realização da dinâmica. Essa maior participação dos estudantes pode ter contribuído positivamente para o aprimoramento de seu desempenho e compreensão do conteúdo, ressaltando a influência benéfica de estratégias de ensino interativas.

Estes resultados, embora sutilmente diferenciados em algumas dimensões, enfatizam a relevância da seleção de estratégias de ensino apropriadas e evidenciam a necessidade de ponderar o nível de envolvimento dos alunos ao conceber atividades pedagógicas. Isso destaca a intrínseca relação entre métodos de ensino e a absorção do conteúdo, bem como a complexidade subjacente à eficácia do processo educativo.

No decorrer da execução do experimento, destacam-se certas ameaças que podem ter desempenhado um papel relevante na influência dos resultados obtidos. Primeiramente, a impossibilidade de aplicar o experimento sem um prévio conhecimento do conteúdo ministrado em sala de aula em virtude da indisponibilidade de datas para aplicação do experimento, que precisou ser adiado, e do andamento convencional da disciplina. As turmas já tinham recebido conhecimento prévio sobre Listas Encadeadas. O conhecimento prévio em relação ao conteúdo abordado pode ter impactado a forma como o experimento foi conduzido, afetando a adaptação das estratégias pedagógicas.

Uma outra eventualidade foi a necessidade de realizar ajustes no Objeto de Aprendizagem. Tal necessidade decorreu do fato de as turmas já terem sido expostas ao

conteúdo, originalmente apresentado por meio da linguagem de programação Java. Entretanto, o Objeto de Aprendizagem concebido por Sousa (2019) foi elaborado utilizando a linguagem de programação C++. Diante deste cenário foi preciso ajustar pontos importantes, especialmente no que se refere à gestão das operações das Listas Encadeadas, por exemplo como seria realizada a liberação de memória após a remoção de um nó da Lista, dado que as duas linguagens adotam abordagens distintas para tratar essa questão.

Essa adaptação introduziu variáveis adicionais que podem ter impactado o desempenho dos alunos que participaram da dinâmica. As modificações no material educacional podem ter repercutido na forma como os estudantes interagiram com o conteúdo, adicionando complexidade à interpretação dos resultados.

Em virtude dessas limitações, é crucial considerar os fatores citados como elementos de relevância na avaliação dos resultados do experimento. Compreender o impacto potencial dessas variáveis adicionais é essencial para uma interpretação rigorosa e abrangente das conclusões do estudo.

6. CONCLUSÃO

Ao longo deste estudo, buscamos objetivos incluindo a revisão do estado da arte sobre Objetos de Aprendizagem na Ciência da Computação, a compreensão das dificuldades dos estudantes no aprendizado de Programação, a validação da utilização de Objetos de Aprendizagem como estratégia de ensino na disciplina de Estrutura de Dados, e a apresentação de propostas para futuros trabalhos que envolvem o desenvolvimento de novas abordagens pedagógicas.

Foi possível analisar e compreender a importância dos Objetos de Aprendizagem como uma valiosa ferramenta na educação contemporânea. Ficou evidente, tanto com base nas pesquisas já realizadas quanto nos resultados do experimento conduzido, que os Objetos de Aprendizagem não devem ser vistos como substitutos dos métodos tradicionais de ensino, mas sim como uma alternativa complementar e eficaz, destinada a enriquecer o processo de aprendizado, servindo como um valioso apoio para o ensino.

As descobertas apontadas nesta pesquisa confirmam que os Objetos de Aprendizagem podem trazer inúmeros benefícios para o fluxo de ensino de Estrutura de Dados especialmente para compreensão de Listas Simplesmente Encadeadas. Eles têm o potencial de tornar o ensino mais dinâmico, interativo e personalizado, proporcionando aos alunos uma experiência de aprendizado mais envolvente. Além disso, permitem que os educadores explorem abordagens inovadoras para as aulas, adaptando-se melhor às necessidades individuais de cada estudante.

Entretanto, é importante ressaltar que, apesar de todos os avanços e benefícios associados aos objetos de aprendizagem, ainda existe uma necessidade premente de investir mais na inovação de novos métodos. Especificamente, na área da Computação, na qual a demanda por recursos de aprendizagem é constante, a escassez de acervos de Objetos de Aprendizagem se torna evidente. Este cenário destaca a importância de direcionar recursos para a criação, adaptação e expansão de Objetos de Aprendizagem, a fim de atender às demandas dos alunos e dos educadores em constante evolução.

Portanto, é possível concluir que os Objetos de Aprendizagem são uma ferramenta valiosa que pode contribuir de forma significativa para a melhoria do ensino e da aprendizagem. No entanto, para maximizar seu potencial e enfrentar os desafios atuais da educação, é imperativo que haja um maior empenho no desenvolvimento de novos métodos e recursos, garantindo que os educadores estejam mais preparados para enfrentar as demandas educacionais do século XXI. Somente assim será possível proporcionar uma educação de qualidade e relevante para todos os estudantes.

Este trabalho teve como foco a apresentar uma abordagem didática destinada a ser empregada em sala de aula como uma ferramenta complementar ao ensino de Estrutura de Dados, especificamente abordando Listas Simples e Dinamicamente Encadeadas. Uma das principais dificuldades encontradas durante a realização deste estudo foi a adaptação do Objeto de Aprendizagem utilizado em sala de aula para integrá-lo ao conteúdo já apresentado à turma.

Como sugestão para futuras pesquisas, destaca-se a possibilidade de expansão do escopo deste trabalho, sendo importante considerar a inclusão de outros conteúdos, como estruturas do tipo Lista Duplamente Encadeadas, para enriquecer ainda mais o material didático. Além disso, é importante explorar a implementação de novos métodos de ensino que possam ser aplicados no apoio à disciplina de Estrutura de Dados, visando a diversificação e aprimoramento das estratégias pedagógicas utilizadas.

REFERÊNCIAS

ANTÓN-RODRÍGUEZ, M.; PÉREZ-JUÁREZ, M. Á.; DÍAZ-PERNAS, FJ; GONZÁLEZ-ORTEGA, D.; MARTÍNEZ-ZARZUELA, M.; AGUIAR-PÉREZ, JM **Uma Experiência de Aprendizagem Baseada em Jogos em Cursos de Desenvolvimento de Aplicações Web.** Ano de publicação: 2020. Disponível em: <https://drops.dagstuhl.de/opus/volltexte/2020/12290/pdf/OASICs-ICPEC-2020-3.pdf>

BACKES, A. **Estrutura de Dados descomplicada em linguagem C.** 1. ed. Rio de Janeiro: Elsevier, 2017.

BROWN, L.; RANELLUCCI, J. **Desafios no ensino de informática para professores de sala de aula tradicional.** Em Anais do 50º Simpósio Técnico ACM sobre Educação em Ciência da Computação (pp. 1527-1527). 2019.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. **Algoritmos - Teoria e Prática.** 2. ed. Rio de Janeiro: Campus. 2002.

CAMPBELL-KELLY, M., ASPRAY, W., ENSMENGER, N.; YOST, J. **Computador: Uma História da Máquina da Informação.** Imprensa Westview. 2014.

CAMPBELL-KELLY, M., GARCIA-SWARTZ, D.; SMITH, MR. **Inovação e comercialização na indústria de TI: explorando o papel das patentes na ciência da computação.** The Economic History Review, 67(4), 1072-1097. 2014

COPELAND, BJ. **Colossus: Os segredos dos computadores decifradores de Bletchley Park.** Imprensa da Universidade de Oxford. 2010.

CORMEN, Thomas H. et al. **Introdução a Algoritmos.** MIT Press, 2009.

GUZDIAL, M. **Projeto de ensino de computação centrado no aluno: pesquisa sobre computação para todos.** Palestras Síntese sobre Informática Centrada no Homem, 10(2), 1-97. 2017.

GUIMARÃES, E.; CARDOSO, PC. **Ensino de informática: tendências e desafios.** Em Anais da Conferência ACM 2017 sobre Inovação e Tecnologia no Ensino de Ciência da Computação (pp. 372-372). 2017

GOODRICH, MICHAEL T., e outros. **"Estruturas de dados e algoritmos em Python."** Wiley, 2013.

GOMES, A.; HENRIQUES, J.; MENDES, A. **Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores.** Educação, Formação; Tecnologias, v. 1, n. 1, p. 93-103, mai. 2008. Disponível em <http://eft.educom.pt>. Acesso em: 18/12/2018.

GROVER, S.; PEA, R. **Pensamento computacional em K-12: Uma revisão do estado do campo.** Pesquisador educacional, 42(1), 38-43. 2013.

HEVNER, AR, MARCH, ST, PARK, J.; RAM, S. **Design Science em Pesquisa de Sistemas de Informação**. MIS Quarterly, 28(1), 75-105. 2004.

HSU, YC, WU, HK E HWANG, FK. **Revisão das tendências dos estudos de aprendizagem móvel: uma meta-análise**. Computers; Education, 59(2), 817-827. 2013.

IEEE - Institute of Electrical and Electronics Engineers. **Learning Object Metadata**. Disponível: <<http://ltsc.ieee.org/wg12/index.html>>. Acesso: 30 fev. 2014. 2005.

ISTE e CSTA. **Computational thinking teacher resources. tion**. Disponível em https://cdn.iste.org/www-root/2020-10/ISTE_CT_Teacher_Resources_2ed.pdf. Acesso em: julho de 2021. 2011.

KITCHENHAM, B. A.; CHARTERS, S. **Guidelines for performing systematic literature reviews in software engineering**. Technical Report EBSE-2007-01, School of Computer Science and Mathematics, Keele University. 2007.

KALLONIATIS, A. **Um exame da eficácia educacional de objetos de aprendizagem para o ensino de Programação Introdutória no ensino médio**. Ano de publicação: 2018. Disponível em: <http://ikee.lib.auth.gr/record/309519/files/GRI-2019-26264.pdf>

KAFAI, YB E BURKE, Q. **A programação de computadores volta à escola**. Phi Delta Kappan, 94(5), 61-65. 2013.

LYYTINEN, K., LOUCOPOULOS, P., MYLOPOULOS, J.; ROBINSON, W. **Desenhar sistemas de informação no contexto social: A abordagem SCOPE**. Imprensa da Universidade de Oxford. 2016.

RODRIGUES, L.; SILVA, C. **Ensino de informática no ensino superior: uma revisão de literatura**. Ciências da Educação, 9(1), 38. 2019.

RIBEIRO, Maria Ivanilse Calderon; PASSOS, Odette Mestrinho. **Um Estudo sobre as Metodologias Ativas Aplicadas ao Processo de Ensino e Aprendizagem na Área de Computação**. Ano de publicação: 2020. Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9252881>

RAGONIS, N.; MARQUES, F. **O uso da robótica educacional para ensinar pensamento computacional: uma revisão sistemática da literatura**. Computadores e Educação, 133, 81-103. 2019.

SMITH, SS; JOHNSON, DM. **Ensino de informática: desafios e oportunidades**. Comunicações do ACM, 61(3), 25-27. 2018.

SOUZA, GM, DORNELES, D., BARBON JR, S., DA SILVA, RP; MEIRA, SR. **Uma revisão sistemática sobre o uso de objetos de aprendizagem no ensino de ciência da computação**. Educação em Ciência da Computação, 30(4), 359-395. 2020.

SEDGEWICK, R.; WAYNE, K. **Algoritmos. 4ª ed**. São Paulo: Editora Pearson Addison-Wesley. 2011.

SOUZA, C.R.; SILVA, N.R. DA; QUEIROZ, R.G. DE; NASCIMENTO, B.R.O. **ITILKnow: um objeto de aprendizagem baseado em jogos para o ensino de ITIL**, In: VIII LACLO 2013, Valdivia, Chile. 2013.

SANTOS, R. P. COSTA, H. A. X. **Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos iniciantes em Computação e Informática**. INFOCOMP, v. 5, n. 1, p. 41-50, 2006. Disponível em: <http://infocomp.dcc.ufla.br/index.php/INFOCOMP/article/view/121>.

SANTOS, R. P. COSTA, H. A. X. **TBC-AED: Um Software Gráfico para Apresentação de Algoritmos e Estruturas de Dados aos Iniciantes em Computação e Informática. I Congresso de Computação do Sul do Mato Grosso (COMPSULMT)**, Rondonópolis, 2005. Disponível em: <https://www.cos.ufrj.br/~rps/pub/completos/2005/COMPSULMT.pdf>. Acesso em: 25 jun.2019.

SOUSA, WP **Computação Desplugada no Apoio ao Ensino de Estrutura de Dados**. Ano de publicação: 2019.

SILVA, João; SOUSA, Maria. **Desafios e Soluções na Educação em Programação: Uma Revisão Sistemática**. Revista Internacional de Educação em Ciência da Computação nas Escolas, São Paulo, v. 5, n. 2, p. 120-135, jun. 2022.

SENTANCE, S.; CSIZMADIA, A. **O impacto da programação na resolução de problemas: uma revisão das evidências**. ACM Transactions on Computing Education (TOCE), 17(1), 1-33. 2017.

TAROUCO, LMR; BULEGON, AM; ÁVILA, BG afirmou em seu artigo "**Objetos de aprendizagem – uso e reuso & intencionalidade pedagógica**" que o uso e a reutilização de objetos de aprendizagem são essenciais para promover a intencionalidade pedagógica. 2021. Disponível em: <https://ieducacao.ceie-br.org/objetos-de-aprendizagem>

TIOSSO, F.; BRUSCHI, S.M.; SOUZA, P.S.L DE; BARBOSA, E.F." **Amnésia: Um Objeto de Aprendizagem para o Ensino de Hierarquia de Memória**", In: 3o CBIE 2014. SBIE. Dourados: Sociedade Brasileira de Computação, 2014.

WING, J. M. **Computational thinking**. Communications of the ACM, 49(3):33–35. 2006.

WONG, Yoke Seng; MAIZATUL HAYATI MOHAMED YATIM. **Jogo de Computador como Ferramenta de Aprendizagem e Ensino de Programação Orientada a Objetos em Instituição de Ensino Superior**. Ano de publicação: 2013. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877042814014554>

APÊNDICE A - PLANO DE AULA**ÁREA DO CONHECIMENTO/TEMA**

Estrutura de Dados: "Listas Dinâmicas Simplesmente Encadeadas"

OBJETIVOS**GERAL:**

- Capacitar o aluno para compreender o funcionamento de estruturas de dados do tipo listas.

ESPECÍFICO:

- Apresentar os conceitos que envolvem estrutura de dados do tipo listas. Seu funcionamento e operações.
- Aplicar a abordagem didática envolvendo os conceitos apresentados no OA.

CONTEÚDO

Estrutura de dados do tipo listas dinâmicas simplesmente encadeadas e suas operações fundamentais.

METODOLOGIA

- Aula expositiva dialogada com explicação detalhada sobre a temática;
- A aula contemplada pelo tema em questão utilizar-se-á de pincel e quadro branco para demonstração dos conceitos e das estruturas.
- Aplicação da abordagem didática em sala de aula, como forma de reforçar os conceitos apresentados, aliando a teoria com a prática.

AVALIAÇÃO

- A avaliação de aprendizagem será verificada mediante aplicação de um questionário para que o estudante ponha em prática os conhecimentos abordados na aula.

REFERÊNCIAS

ASCÊNCIO, A.F.G.; CAMPOS, E.A.V Fundamentos da Programação de Computadores. São Paulo: Editora Pearson Prentice Hall, 2007.

BACKES, A. Estrutura de Dados descomplicada em linguagem C. I. ed. Rio de Janeiro: Elsevier, 2017.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. Algoritmos - Teoria e Prática. 2. ed. Rio de Janeiro: Campus. 2002

APÊNDICE B - QUESTIONÁRIO UTILIZADO NO EXPERIMENTO

Questionário construído para avaliar o desempenho e a absorção dos alunos com relação ao conteúdo exposto no plano de aula (Ver Apêndice A):

Questão 1: O que é uma lista simplesmente encadeada?

- a) Uma estrutura de dados que armazena elementos em pares chave-valor.
- b) Uma estrutura de dados que organiza elementos em uma sequência linear, onde cada elemento é representado por um nó com um ponteiro para o próximo nó.
- c) Uma estrutura de dados que organiza elementos em uma árvore binária.
- d) Uma estrutura de dados que armazena elementos em uma matriz bidimensional.

Questão 2: Como é chamada a operação que adiciona um elemento no início de uma lista simplesmente encadeada?

- a) Pop Front
- b) Push Middle
- c) Pop Back
- d) Push Front

Questão 3: Qual a principal vantagem das listas simplesmente encadeadas em relação às operações de inserção e remoção?

- a) Acesso aleatório.
- b) Uso eficiente de memória.
- c) Facilidade de busca.
- d) Estrutura estática.

Questão 4: Como é chamada a operação que remove o elemento no final de uma lista simplesmente encadeada?

- a) Push middle
- b) pop Back
- c) Pop Specific Position
- d) Push Front

Questão 5: Qual a desvantagem das listas simplesmente encadeadas em relação ao acesso aleatório?

- a) Eficiência na busca.
- b) Uso eficiente de memória.
- c) Inserção no início.
- d) Ineficiência na busca.

Questão 6: Qual é o nome dado ao último nó de uma lista simplesmente encadeada?

- a) Início
- b) Calda
- c) Atual
- d) Cabeça

Questão 7: Qual é a operação que permite encontrar um elemento em uma lista simplesmente encadeada?

- a) Busca
- b) Inserção
- c) Remoção
- d) Criação

Questão 8: Qual é a estrutura de dados que possui apenas uma direção de conexão entre nós?

- a) Lista Dinâmica Duplamente Encadeada
- b) Árvore Binária
- c) Fila
- d) Lista Dinâmica Simplesmente Encadeada

Questão 9: Inserção no Início.

- a) Descreva o processo de inserção de um novo nó no início de uma lista simplesmente encadeada. Quais são as etapas envolvidas? Como você ajustaria os ponteiros para garantir que a nova inserção fosse bem sucedida?

Questão 10: Inserção no Fim.

- a) Explique como a operação de inserção no final de uma lista simplesmente encadeada é realizada. Quais são as considerações a serem feitas ao realizar essa operação?

Questão 11: Remoção no Início.

- a) Discuta a remoção de um nó no início de uma lista simplesmente encadeada. Quais são os passos envolvidos nessa operação?

Questão 12: Remoção no Fim.

- a) Descreva o processo de remoção de um nó no final de uma lista simplesmente encadeada. Quais são os desafios ao realizar essa operação e como você lidaria com eles para garantir que o nó correto seja removido?

Questão 13: Procurar um Nó.

- a) Como você implementaria uma função para procurar um nó específico em uma lista simplesmente encadeada? Descreva o algoritmo que você usaria e como ele funcionaria. Que cuidados você deve ter ao procurar um nó na lista?