

Fórmula Tree: Um Jogo Sérió para o Ensino e Aprendizagem de Estrutura de Dados

**Arthur Monteiro Pereira¹, Filipe Brinati Furtado¹, Paulo Victor de Magalhães Rozatto ¹,
Pedro Henrique Filgueiras dos Santos Oliveira¹**

¹Departamento de Ciência da Computação – Universidade Federal de Juiz de Fora (UFJF)
Caixa Postal 20010 – CEP 36036-900– Juiz de Fora – MG – Brazil

Abstract.

Resumo.

1. Introdução

2. Uso de jogos no ensino e aprendizagem de Estrutura de Dados

Com o desenvolver do curso de ciência da computação, docentes de suas disciplinas são cada vez mais desafiados a proporcionar uma qualidade de ensino suficiente para o desenvolver dos alunos [Qian and Lehman 2017]. Além disso, as áreas de conhecimento de ciência da computação costumam consistir de fundamentos distantes do senso comum, ou seja, para aprendê-las é necessário um nível de abstração que costuma ser difícil de compreender por alunos iniciantes no conteúdo [Rojas-Salazar 2022].

Partindo desse pressuposto, é evidente que a disciplina de Estrutura de Dados também compartilha de tais características. Assim, tal disciplina que faz a ponte inicial entre os fundamentos de computação, apresentando conceitos como listas, pilhas, filas, árvores e pesquisas nessas, para disciplinas mais complexas futuramente apresentadas nos cursos de computação possui tamanha importância em seu entendimento [Lawrence 2004]. Como consequência, alunos que não conseguirem fazer um bom proveito dessa matéria podem apresentar dificuldades no desenvolver do curso.

No entanto, em muitos dos casos alunos costumam apresentar desinteresse nos métodos clássicos utilizados para o ensino e aprendizagem de Estrutura de Dados, apresentando resultados sub-par na completção das tarefas requisitados a eles [Lawrence 2004]. Isso fica explícito na publicação [da Silva 2021] que apresenta uma porcentagem de repetição na disciplina de Estrutura de Dados acima de 40% na Universidade Federal Rural do Semi-Árido, Campus Pau dos Ferros - RN (UFERSAPDF), com casos de acima de 70% durante a pandemia do SARS-CoV-2 (COVID-19).

Diante de tais desafios, diferentes métodos têm sido desenvolvidos para auxiliar o ensino e aprendizagem de Estrutura de Dados ao longo dos anos, dentre eles simulações, ferramentas de visualização e jogos sérios [Lawrence 2004].

Assim como jogos sérios em geral, a introdução desses no ensino e aprendizagem de Estrutura de Dados é uma pesquisa recente tornando difícil quantificar a contribuição desses no ensino. Isso é evidenciado no mapeamento realizado na tese [Rojas-Salazar 2022], a qual apenas encontrou 17 jogos sérios utilizados para auxiliar o ensino da disciplina.

A partir desse mapeamento, é notável que as pesquisas na área preferem fazer jogos sérios para auxiliar a parte inicial da matéria, ensinando em sua maioria as diferentes maneiras de se implementar listas e algoritmos recursivos, enquanto os conceitos de árvores e suas pesquisas são pouco priorizados. No mais, quanto aos elementos dos jogos, foi ressaltado que a maioria dos jogos prefere utilizar de ambientes e mecânicas de jogos para integrar os conceitos de Estruturas de Dados e complementar com elementos de competição. Poucos foram os casos que optaram em introduzir conceitos idênticos ao apresentados nos livros.

Quanto a avaliação do uso de jogos sérios, pesquisadores buscando por seus métodos compreender os impactos evidenciais e a eficiência de tal aplicação. No entanto, para isso é notável na literatura que para coleta dos resultados ainda é presente o uso de ferramentas informais. Com isso, é difícil avaliar um padrão no desenvolvimento de jogos sérios que garanta resultado, já que a aplicação é realizada em di-

ferentes contextos. Todavia, diversas ferramentas de avaliação tem começado a ganhar cada vez mais espaço na área de avaliação de jogos sérios, como apresentado em [Petri and Gresse von Wangenheim 2017], buscando por meio dessas, criar um padrão de avaliação para futuras pesquisas que fazem o uso de jogos sérios no ensino.

Concluindo, o uso de jogos sérios no ensino e aprendizagem de Estrutura de Dados tem cada vez mais ganhado espaço na área de pesquisa. Optando em desenvolver jogos para auxiliar as etapas iniciais de ensino dessa disciplina, a qual é essencial para o aprendizado do resto de tal. Porém, a falta de um método comum de avaliação dificulta uma conclusão dos resultados de tal aplicação, mesmo que diferentes métodos de avaliação de jogos já estejam presentes no cenário de pesquisa.

3. Descrição do jogo

4. Trabalhos Relacionados

O uso de desenvolvimento de jogos sérios para ensino de Estrutura de Dados tem ampla cobertura na literatura. Entre diferentes trabalhos desenvolvidos, diversas estruturas são abordadas. No entanto, nenhum trabalho relacionado citado nesse trabalho aborda estrutura de árvores, assunto que é o foco central de Formula Tree.

Uma das primeiras coisas que um estudante de Estrutura de dados deve aprender é a estrutura da lista encadeada. Uma lista encadeada é composta por nós, que têm ponteiros apontando para o próximo nó da lista, ao passo que o último nó aponta para um endereço nulo. Em adição a isso, esses nós têm um ponteiro chamado *head*, que aponta para o primeiro elemento da lista.

Diante disso, a lista se oferece como uma opção promissora para auxílio de novatos em dificuldades no curso. Um trabalho que tem o foco de auxiliar o ensino de listas encadeadas é [Kannappan et al. 2019], que desenvolve um jogo do gênero *puzzle* chamado La Petite Fee Cosmo.

Esse jogo, desenvolvido com o uso da *engine* da Unity, se baseia no uso de falha produtiva [Kapur 2008] para facilitar o aprendizado dos jogadores. Portanto, a intenção é ter um jogo desafiador o suficiente para estimular a evolução, mas que não seja difícil demais a ponto de tornar-se frustrante e desmotivador.

A ideia geral de La Petite Fee Cosmo é fazer com que a personagem Cosmo construa pontes, resolvendo diferentes quebra-cabeças. As pontes são uma abstração para uma lista encadeada. Uma pedra que compõe a ponte representa um nó da lista.

O jogo é dividido em 3 capítulos: o primeiro consiste em desafiar o jogador a colocar o maior ou menor elemento da lista na última ou primeira posição, introduzindo o estudante à noção de troca de posição de elementos.

O segundo capítulo é uma extensão do primeiro, tendo como principal tema de desafio a reordenação completa da lista (*i.e.* elementos em ordem crescente ou ordem decrescente). Neste capítulo, o entendimento acerca do ponteiro *head* também passa a ser mais exigido.

Por fim, o terceiro capítulo busca desafiar de maneira mais geral o jogador. Neste, o quebra-cabeça consiste em posicionar elementos de uma ponte inferior em diferen-

tes posições na ponte superior, exigindo do jogador entendimento mais avançado de manipulação dos nós, interação entre diferentes listas e assim por diante.

Os autores não realizaram dentro desse trabalho nenhum tipo de experimento que possa atestar a qualidade do produto desenvolvido, mas isso consta como uma das intenções de trabalhos futuros. Além disso, outra ideia é expandir o escopo do jogo, implementando desafios para outras estruturas, como pilhas, filas, árvores e etc.

Já em [Telinski Wiedermann Agner et al. 2021], o uso de jogos sérios é o usado para o ensino de estruturas de dados, mas não para que o aprendizado ocorra apenas com o ato de jogar o jogo pronto, mas que o processo de desenvolvimento, seguindo o contexto de Aprendizagem Baseada em Projetos (PBL - *Project Based Learning*), cumpra esse papel. O projeto foi desenvolvido como uma atividade proposta na disciplina de Algoritmos e Estruturas de Dados I (AEDI) do curso de Bacharelado em Ciência da Computação da Universidade Estadual do Centro-Oeste no ano de 2020.

Este processo de desenvolvimento contém além das fases de modelagem, em que se tem contato com os diagramas da linguagem de modelagem padrão da indústria de Engenharia de Software atualmente, a UML (Unified Modeling Language), a fase de escolha de qual estrutura de dados usar, fase muito importante em que se exercita a capacidade de identificar qual a estrutura adequada para o problema e a fase de implementação, onde se tem o foco de praticar de fato a programação de tais estruturas.

Um dos projetos apresentados utiliza o conceito de FIFO (*First In, First Out*) para implementar uma versão do jogo 2048 que possui um tabuleiro 4x4 com peças vinculadas a um valor numérico múltiplo de 2, em que o objetivo é combinar essas peças para alcançar o valor de 2048. O projeto usou as setas do teclado para controlar esse *input*.

Ao fim, chegou-se a conclusão que esta gamificação do ensino é muito importante para melhorar a taxa de retenção do conhecimento por parte dos alunos possibilitando experimentar os conceitos em problemas reais. Entretanto, essa abordagem também necessita de planejamento, exigindo novas experiências não apenas dos alunos, mas também dos professores.

Voltando ao contexto de um aprendizado mais tradicional baseado em jogos sérios, em [Ramle et al. 2019], é abordado um assunto específico dentro da disciplina de estrutura de dados: a estrutura da pilha.

O trabalho citado busca fazer isso criando um ciclo em que o usuário aprecia o jogo, levando a um novo comportamento (*i.e.*, jogar de novo), enquanto recebe *feedbacks* do sistema. Com isso, o usuário desenvolve um aprendizado procedural, sendo assim capaz de estruturar um raciocínio mais algorítmico para resolução de problemas de pilhas.

Pilhas são estruturas de dados cujos elementos são ordenados por meio do LIFO (*Last In, First Out*). Nela, há duas operações possíveis: *push*, que insere um elemento no topo da pilha, e *pop*, que exclui o elemento no topo da pilha. O jogo produzido busca auxiliar estudantes na abstração referente ao procedimento citado.

Nesse contexto, o jogador responde a perguntas para construir uma pilha de pedras. A cada acerto, pontos são conquistados. Mas a cada erro, pontos de vida, que iniciam com valor de 5, são perdidos. Ao zerar os pontos de vida ou encerrar o tempo, o jogo se encerra.

O jogo, chamado Train to Kluang, não possui *link* para *download* disponível em [Ramle et al. 2019]. Este foi desenvolvido com o uso do Adobe Photoshop para *assets* gráficos e com o uso do Adobe Flash para a integração de todos os elementos da aplicação.

Para uma avaliação, foram realizados experimentos, em que uma amostra de 29 estudantes de computação que haviam cursado a disciplina de Estrutura de Dados no semestre anterior foram selecionados. Jogadores responderam a uma série de perguntas antes e depois de jogar Train to Kluang. Para um total de 8 tipos de pergunta, em 7 deles o número de alunos acertando aumentou após jogar o jogo.

Por fim, os estudantes responderam a um questionário com escala *Likert* para avaliar a usabilidade do jogo, além de enviar sugestões. Avaliações de estudantes foram em grande parte positivas, mas houve sugestões de melhoria em relação à comunicação visual do projeto.

Outra abordagem para o ensino da estrutura de dados de pilha através de um jogo é fornecido por [Dicheva and Hodge 2018]. Os autores se propõem a abordar três objetivos associados ao ensino de pilhas: o entendimento conceitual teórico de tal estrutura, a aprendizagem de como aplicar o entendimento conceitual em problemas reais e a aprendizagem de implementar essas estruturas em código.

Para cumprir suas propostas, [Dicheva and Hodge 2018] criaram o jogo nomeado The Stack Game, no qual o usuário controla um robô que sofreu um acidente, teve sua nave destruída e agora tem que passar por diversos desafios para conseguir chegar em seu lar. O The Stack Game é dividido em três partes, cada uma explorando um objetivo de aprendizagem.

Na primeira parte do jogo, o objetivo é o entendimento conceitual da estrutura de pilha. O mecanismo de ensino utilizado são problemas baseados no quebra-cabeça Torre de Hanoi: o jogador é dado uma pilha de blocos coloridos em uma certa ordem e tem que controlar seu robô para deixá-la em outra configuração determinada. Cumprido o objetivo, a porta da fase se abre e o jogador consegue passar para próxima, sendo que a primeira parte tem quatro fases de dificuldade progressiva.

Na segunda parte do jogo, o objetivo de aprendizagem é aprender a aplicar o conceito de pilhas em problemas reais. Durante as fases dessa parte, o jogador tem que resolver os seguintes problemas: converter uma expressão aritmética escrita na notação infixa para a notação posfixa, avaliar expressões em notação posfixa e em notação infixa. Antes de cada fase o jogador é instruído sobre a tarefa a ser realizada e os operadores disponíveis para se utilizar.

Por fim, na terceira parte do jogo, o objetivo é a aprendizagem de como se implementar pilhas. Nessa parte há apenas uma fase e o objetivo é acender quatro lâmpadas inicialmente apagadas. Para se acender as lâmpadas, é necessário entrar com o código correto de quatro funções ligadas a operações em pilhas: empilhar(), desempilhar(), buscar(), estaVazia(). Cada método corretamente implementado acende uma lâmpada e ao se acender todas, a fase é concluída e o jogo é finalizado.

A avaliação do The Stack Game foi realizada em dois semestres letivos consecutivos com os alunos matriculados no curso de estrutura de dados da Southern Liberal

Arts College. O jogo foi usado como atividade regular durante as aulas e todos os alunos participaram, logo não houve grupo de controle. Os participantes foram submetidos a um teste antes de terem contato com o jogo e outro teste similar após contato com o jogo. Os resultados encontrados apontaram um resultado significativo na melhora de desempenho no segundo teste realizado e na avaliação qualitativa também realizada, a percepção dos participantes foi de que o The Stack Game ajudou na motivação para aprender o conteúdo e a clarificar os conceitos de programação.

Diferentemente dos trabalhos citados, [Su et al. 2021] cria um jogo que aborda diversos tipos de estruturas de dados, ao invés de focar em um único tipo. O projeto é nomeado como deCode.

A proposta desse projeto foi desenvolver uma aplicação que unisse conceitos de jogos sérios com uma aprendizagem baseada em visualização. Estudos mostram que a abordagem de visualização sozinha não é suficiente para o aprendizado de algoritmos [Hundhausen et al. 2002]. Por esse motivo, [Su et al. 2021] buscam unir a visualização com a interação gerada pelos jogos.

A principal ideia trazida por deCode é a de usar metáforas para facilitar a compreensão de conceitos de difícil leitura por parte de iniciantes. Dessa forma, o jogo tem diversas fases que contêm quebra-cabeças envolvendo diferentes partes da disciplina de estrutura de dados.

Em todas essas, carros representam dados ou nós a serem inseridos em uma vaga no estacionamento, que por sua vez é análoga a um endereçamento de memória.

A primeira fase tem um quebra-cabeça envolvendo estrutura de vetores (*arrays*). Em sequência, as fases seguintes exploram, em ordem, o conceito de listas, filas e pilhas.

Reorganizando os carros nas vagas de estacionamento, os jogadores podem abstrair conceitos complexos de estrutura de dados.

Nesse caso, a aplicação foi executada em um servidor *web* e esteve disponível para voluntários testarem o jogo. Com isso, eles puderam responder a questionários *Likert* para avaliar a experiência que tiveram. Trinta alunos participaram do experimento preenchendo o formulário e o *feedback* em geral foi positivo. No entanto, como se tratava de voluntários, isso pode ter causado algum tipo de viés no perfil de estudantes participantes.

5. Análise comparativa

6. Considerações finais

Referências

- da Silva, F. L. A. (2021). Um relato de experiência do uso de metodologias ativas para o ensino remoto de estruturas de dados em tempos de pandemia an experience report of the use of active methodologies for the remote teaching of data structures in times of pandemics. *Brazilian Journal of Development*, 7(7):70453–70491.
- Dicheva, D. and Hodge, A. (2018). Active learning through game play in a data structures course. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, pages 834–839.

- Hundhausen, C. D., Douglas, S. A., and Stasko, J. T. (2002). A meta-study of algorithm visualization effectiveness. *Journal of Visual Languages & Computing*, 13(3):259–290.
- Kannappan, V. T., Fernando, O. N. N., Chattopadhyay, A., Tan, X., Hong, J. Y. J., Seah, H. S., and Lye, H. E. (2019). La petite fee cosmo: Learning data structures through game-based learning. In *2019 International Conference on Cyberworlds (CW)*, pages 207–210.
- Kapur, M. (2008). Productive failure. *Cognition and instruction*, 26(3):379–424.
- Lawrence, R. (2004). Teaching data structures using competitive games. *IEEE Transactions on Education*, 47(4):459–466.
- Petri, G. and Gresse von Wangenheim, C. (2017). How games for computing education are evaluated? a systematic literature review. *Computers Education*, 107:68–90.
- Qian, Y. and Lehman, J. (2017). Students’ misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education (TOCE)*, 18(1):1–24.
- Ramle, R., Nathan, S. S., Berahim, M., et al. (2019). Digital game based learning of stack data structure using question prompts.
- Rojas-Salazar, A. J. (2022). Game-based learning of data structures based on analogies: Learning gains and intrinsic motivation in higher education environments.
- Su, S., Zhang, E., Denny, P., and Giacaman, N. (2021). A game-based approach for teaching algorithms and data structures using visualizations. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education, SIGCSE ’21*, page 1128–1134, New York, NY, USA. Association for Computing Machinery.
- Telinski Wiedermann Agner, L., Ferreira Thomen, M. A., Utzig, G. M., and Wisniewski Soares, I. (2021). Educação em algoritmos e estruturas de dados no contexto do ensino baseado em jogos digitais. *TE & ET*.