

# QS GAME: UM JOGO PARA ENSINO DO ALGORITMO DE ORDENAÇÃO QUICKSORT

Marieli Aparecida Ferreira Thomen<sup>1</sup>; Inali Wisniewski Soares<sup>2</sup>; Luciane Telinski Wiedermann Agner<sup>3</sup>

## Resumo

Em Ciência da Computação a aprendizagem dos métodos de ordenação é importante para que o aluno conheça as diversas formas de classificar e organizar dados. O algoritmo de ordenação QuickSort destaca-se por ser um dos métodos com melhor desempenho na maioria dos casos. Entretanto, o estudo de seu funcionamento é considerado complexo para compreender pela maioria dos alunos. A gamificação aplicada a educação consiste na utilização de jogos sérios para auxiliar no ensino dos conteúdos de maneira lúdica e dinâmica. Este artigo apresenta um jogo que visa facilitar e motivar a aprendizagem desse método de ordenação, denominado QS Game. O jogo possui uma interface gráfica atrativa e utiliza questões organizadas em uma sequência semelhante às etapas de funcionamento do QuickSort e que permite, dessa forma, que o jogador analise e intérprete passo a passo do algoritmo. Desse modo, o jogador pode atuar como protagonista no processo de aprendizagem, interagindo com a ferramenta para ordenar um conjunto de elementos. O jogo também oferece um tutorial disponibilizado por meio de um vídeo explicativo que demonstra o funcionamento do algoritmo.

**Palavras-chave:** gamificação; ciência da computação; algoritmos de ordenação.

## Abstract

In Computer Science, learning sorting methods is important so that the student knows the different ways of classifying and organizing data. The QuickSort algorithm stands out for being one of the best performing methods in most cases. However, the study of its operation is considered complex to understand by most students. Gamification applied to education consists of using serious games to help teach content in a playful and dynamic way. This article presents a game that aims to facilitate and motivate the learning of this sorting method, called QS Game. The game has an attractive graphical interface and uses questions organized in a sequence similar to the QuickSort operating steps, which allows the player to analyze and interpret the algorithm step by step. In this way, the player can act as a protagonist in the learning process, interacting with the tool to order a set of elements. The game also offers a tutorial made available through an explanatory video that demonstrates how the algorithm works.

**Keywords:** gamification; computer science; sorting algorithms.

## 1 Introdução

Em cursos de Ciência da Computação, disciplinas de Algoritmos e Estruturas de Dados contemplam o ensino da organização, manipulação e uso dos dados. Nesse contexto, métodos

---

<sup>1</sup> Bacharel em Ciência da Computação pela Universidade Estadual do Centro-Oeste-UNICENTRO. E-mail: mari\_jawwad@hotmail.com.

<sup>2</sup> Doutora em Engenharia Elétrica e Informática Industrial pela Universidade Tecnológica Federal do Paraná-UTFPR; professora do Departamento de Ciência da Computação da Universidade Estadual do Centro-Oeste-UNICENTRO. E-mail: inali@unicentro.br.

<sup>3</sup> Doutora em Engenharia Elétrica e Informática Industrial pela Universidade Tecnológica Federal do Paraná-UTFPR; professora do Departamento de Ciência da Computação da Universidade Estadual do Centro-Oeste-UNICENTRO. E-mail: lagner@unicentro.br.

de ordenação possibilitam organizar e classificar um conjunto de dados armazenados na memória do computador. Uma das formas mais conhecidas de classificação é a ordenação “por troca”, que consiste em efetuar a classificação a partir de comparações sucessivas de pares de elementos, trocando-os de posição se necessário, com base em requisitos pré definidos. A principal diferença entre os diversos métodos de ordenação por troca é a estratégia utilizada na seleção dos pares de elementos a serem avaliados (TENENBAUM *et al.* 1995).

O QuickSort, um algoritmo de ordenação baseado em troca, é um dos métodos de classificação mais adotados por ser um dos mais rápidos e eficientes na grande maioria dos casos (Piva Junior *et al.* 2014). O funcionamento do QuickSort é baseado em um elemento denominado “pivô”, que pode ser definido a partir da média aritmética entre a posição do primeiro e do último elemento do vetor.

Desse modo, o pivô é o elemento do centro do vetor. Todos os elementos que antecedem o pivô são classificados como parte esquerda do conjunto a ser ordenado e os que o sucedem, como parte de direita. O objetivo é deixar os elementos menores na parte esquerda e os maiores na direita. Esse processo de particionamento é repetido para os segmentos de esquerda e de direita sucessivamente até que o vetor esteja devidamente ordenado (Piva Junior *et al.* 2014; Villas *et al.* 1993). Entretanto, o aprendizado do funcionamento desse método de ordenação é considerado complexo pelos alunos, pois envolve lógica e programação.

A gamificação consiste no uso de jogos sérios em áreas como educação e saúde, por exemplo. No contexto do ensino, visa facilitar o aprendizado e melhorar o desempenho dos alunos, despertando maior interesse, motivação e engajamento. O uso da gamificação possibilita abordar conteúdos de forma dinâmica e interativa, impulsionando a criatividade, a autonomia e a motivação dos alunos (Kosa *et al.* 2016; Kapp, 2012).

Neste contexto, este artigo detalha uma ferramenta pedagógica voltada para o ensino do algoritmo de ordenação QuickSort (thomen *et al.* 2021). A ferramenta é baseada em gamificação e oferece uma proposta atrativa e que possibilita ao aluno desempenhar um papel ativo na construção do conhecimento a respeito do tópico abordado.

## **2 Referencial Teórico**

Esta seção apresenta os principais conceitos utilizados no desenvolvimento do jogo QS Game, sendo eles: gamificação e métodos de ordenação.

## 2.1 Gamificação

No contexto de educação, a gamificação engloba o uso de jogos visando tornar o processo de ensino mais atrativo, bem como despertar a motivação e a curiosidade, além de facilitar o entendimento dos conceitos abordados (LEE e HAMMER, 2011).

Os elementos presentes nos jogos como competição, desafio e recompensa, aumentam o interesse e incentivam a participação dos alunos, além de possibilitar reforçar conhecimentos obtidos. Desse modo, a educação pode se beneficiar fortemente do uso de jogos, tornando o processo de aprendizagem mais dinâmico, interativo e divertido (Kosa *et al.* 2016). Em disciplinas que envolvem temas considerados complexos e de difícil entendimento pela maioria dos alunos, como lógica por exemplo, o uso de ferramentas que auxiliem no processo de ensino, torna-se ainda mais importante (Bigolin *et al.* 2020; Dicheva e Hodge, 2018).

## 2.2 Métodos de Ordenação

No contexto de Ciência da Computação, a tarefa de ordenar corresponde ao processo de rearranjar um conjunto de dados em uma ordem específica, previamente determinada. Assim, métodos de ordenação visam a classificação dos dados com base em um algoritmo de ordenação. O algoritmo de ordenação denominado QuickSort é um dos métodos mais utilizados para classificação de dados (Piva Junior *et al.* 2014; Villas *et al.* 1993).

### 2.2.1. O algoritmo de ordenação QuickSort

QuickSort é um método de ordenação baseado em trocas da posição de elementos. É um dos métodos de ordenação mais utilizados, por apresentar melhor desempenho para uma grande diversidade de situações (TENENBAUM *et al.* 1995).

Considerando que o conjunto de elementos a serem ordenados está armazenado em um vetor, o funcionamento do QuickSort tem como base a seleção de um elemento chamado pivô. O pivô pode ser um elemento localizado no início, meio ou fim do vetor. A partir desse pivô, ocorre a partição do vetor, onde os valores da parte esquerda deverão ser menores ou iguais ao pivô e os valores da parte direita maiores ou iguais ao mesmo. Enquanto esse padrão não é atingido, os valores da esquerda são trocados com valores da direita. Isso ocorre até que todos os valores já estejam posicionados em suas posições corretas, com base no valor do pivô selecionado.

O algoritmo realiza chamadas recursivas das porções esquerda e direita, selecionando um novo pivô e particionando o vetor para fazer novas trocas, de forma recursiva, até que todos os

elementos do vetor estejam devidamente ordenados. Na literatura, diversas versões de algoritmos de implementação do QuickSort são apresentadas (Piva Junior et al, 2014; Tenenbaum et al, 1995). A versão utilizada como base para o desenvolvimento do jogo QS Game foi o algoritmo apresentado por Villas *et al.* (1993).

### 3 QS Game

O QS Game é um jogo que visa auxiliar no ensino do método de ordenação QuickSort. O jogo consiste em apresentar um vetor desordenado ao seu jogador, representado por blocos coloridos que contêm valores a serem ordenados. Perguntas são exibidas, relacionadas com a ordem lógica de solução do problema utilizando o algoritmo QuickSort. Durante o processo, dicas serão apresentadas ao jogador para auxiliar no processo de ordenação.

Como a ordenação utilizando o QuickSort é baseada em trocas, no momento do jogo em que deverão ocorrer trocas, o usuário será alertado por meio das perguntas e deverá selecionar os blocos que representam os valores a serem trocados, desse modo, representando e simulando a execução do algoritmo.

As regras do jogo desenvolvido são as seguintes: i) o jogador inicia o jogo com três vidas; ii) A cada resposta incorreta o jogador perde uma vida; e iii) o jogador tem a chance de tentar responder uma mesma questão do jogo mais de uma vez, até que acerte ou perca todas as vidas. Quando o jogador perde todas as vidas precisa reiniciar o jogo.

#### 3.1 Modelagem do jogo QS Game

A modelagem de casos de uso do jogo foi realizada visando descrever as etapas de ações que podem ser realizadas no jogo implementado. O Diagrama de Casos de Uso, apresentado na Figura 1, mostra os requisitos da ferramenta desenvolvida, tal modelagem possibilita melhor compreensão do domínio da aplicação. O Diagrama de Casos de Uso é um modelo da linguagem de modelagem UML (*Unified Modeling Language*), que possibilita representar o comportamento de um sistema, especificando os atores e suas interações com o software (Sommerville, 2019).

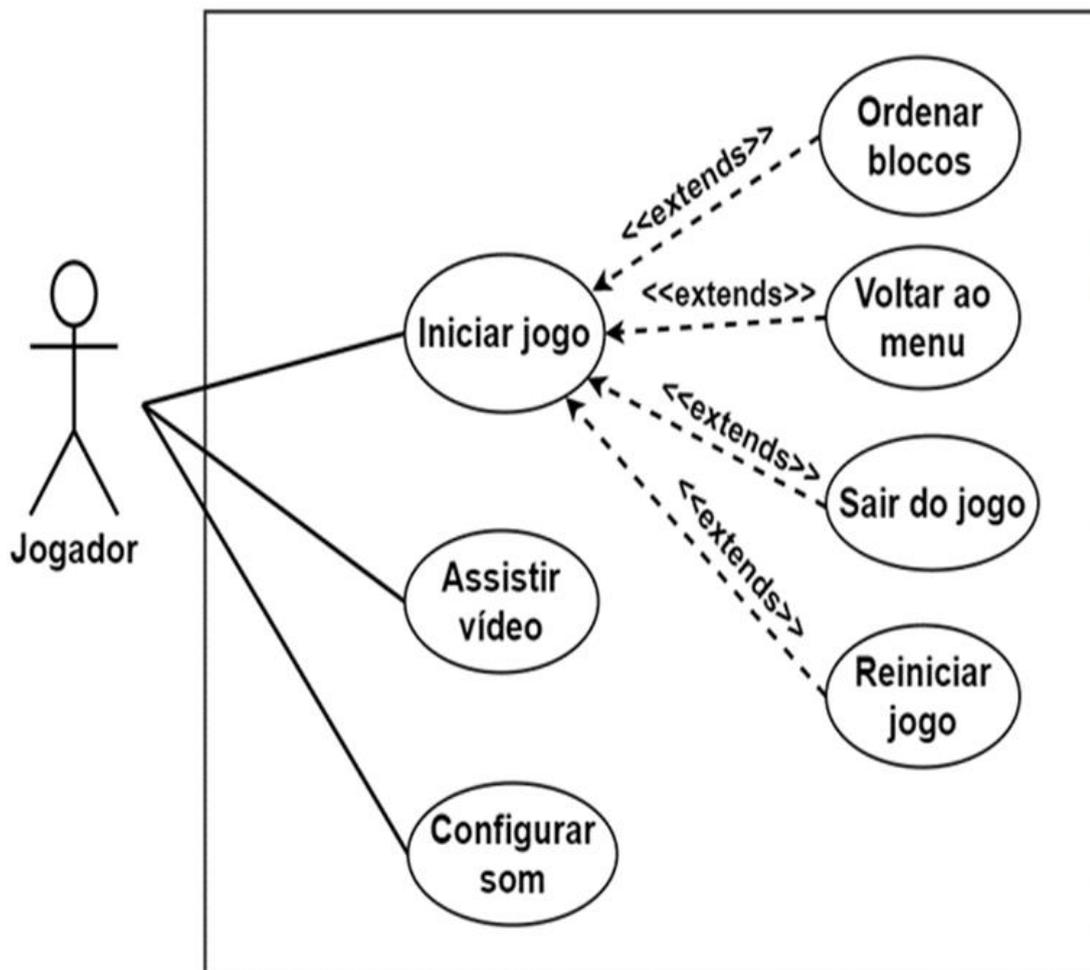
O caso de uso “Assistir vídeo” descreve as etapas percorridas por um jogador para assistir ao vídeo que apresenta um exemplo do funcionamento do algoritmo do QuickSort. Por sua vez, o caso de uso “Iniciar jogo” apresenta as etapas percorridas por um jogador para iniciar e/ou finalizar uma partida no jogo, ou ainda, reiniciar o jogo.

### 3.2 Interface gráfica do jogo

O QS Game é composto por duas telas principais, representadas nas Figuras 2 e 3. A tela apresentada na Figura 2 representa a primeira tela de menu que aparece ao iniciar o jogo. O menu possui quatro botões interativos, o botão X é o índice que permite sair do jogo, a faixa de vídeo possibilita o acesso ao vídeo explicativo, o botão Play inicia o jogo e o índice com o som serve para configurar a trilha sonora do jogo.

A Figura 3 exemplifica o padrão da interface utilizada do jogo, composta pelos elementos gráficos definidos. Foram criados quatro botões interativos, sendo eles: i) a seta na lateral superior direita que serve para voltar ao menu principal; ii) o ponto de interrogação para que o jogador possa visualizar novamente a questão a ser respondida; iii) o botão de confirmação para validar a resposta; e iv) a seta rotacionando no sentido horário que permite reiniciar o jogo. Os três corações definidos no canto superior esquerdo representam as vidas do jogador. Também, os índices de posição de esquerda, pivô e direita são representados nesta tela.

**Figura 1** - Diagrama de Casos de Uso do jogo QS Game



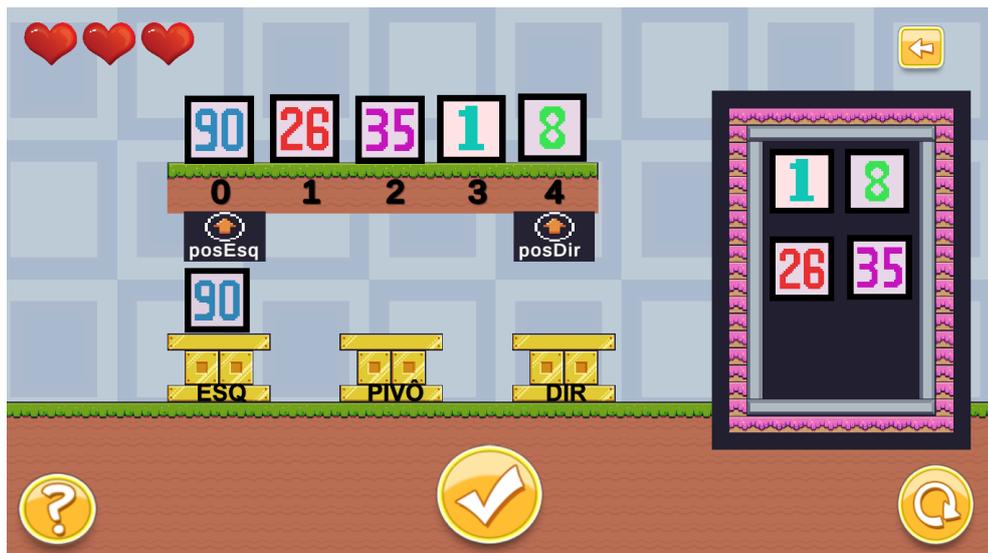
Fonte: Elaborada pelos autores.

**Figura 2 - Menu principal**



Fonte: Elaborada pelos autores.

**Figura 3 - Cenário do jogo QS Game**



Fonte: Elaborada pelos autores.

Para possibilitar a interação do usuário com o jogo, foram criados três suportes para compor a tela do jogo, possibilitando ao jogador posicionar sobre os respectivos blocos os valores estratégicos para a solução do problema naquele momento. As alocações podem ser realizadas pelos usuários utilizando o mouse, de modo a arrastar os blocos dispostos na lateral direita da tela, de acordo com a lógica solicitada naquele momento.

### 3.3 Questões e mensagens

As questões definidas para o QS Game foram representadas conforme a ordem de aparição destas no jogo e nominadas como Qx, onde x representa o número da questão. Desse

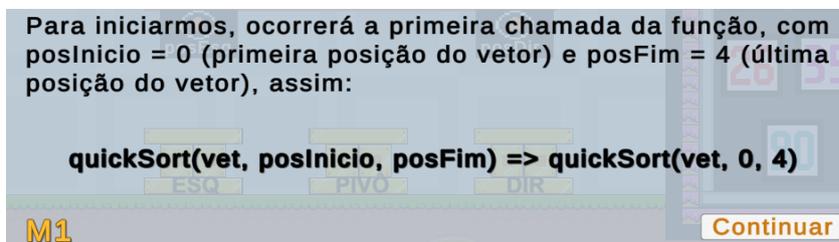
modo, cada questão possui um número que a identifica com base na sequência do fluxo de execução do jogo. O plano de fundo das telas das questões foi definido como branco.

O jogo também apresenta mensagens, para auxiliar os jogadores durante o jogo. Essas mensagens são nominadas por  $Mx$ , onde  $x$  é um número que representa a sequência de apresentação. O plano de fundo para as mensagens foi definido como azul claro. O QS Game contempla 4 mensagens definidas para apresentar ao jogador o contexto do jogo em determinado momento. Desse modo, o objetivo principal das mensagens é mostrar ao jogador o ponto da execução do algoritmo QuickSort.

Uma mensagem é composta por uma introdução que define o contexto da fase em que o jogo se encontra, e um trecho do código que está sendo executado no momento. O botão “Continuar” possibilita ao jogador fechar o quadro com a mensagem e prosseguir para as próximas questões do jogo. A Figura 4 traz como exemplo a mensagem M1 do QS Game, responsável por introduzir a primeira chamada da função para iniciar a ordenação dos elementos do vetor.

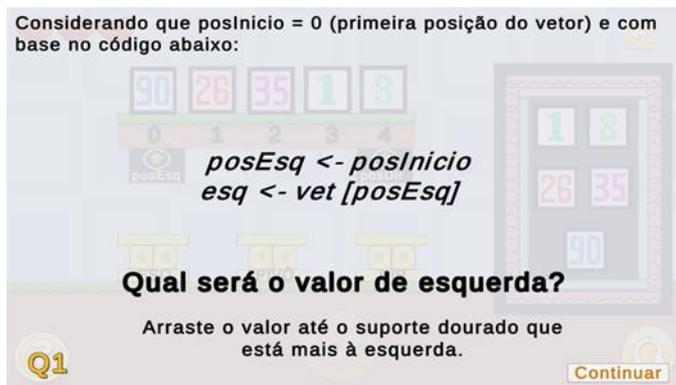
Cada fase do jogo está relacionada com uma questão, composta por: i) introdução, definindo o contexto do questionamento; ii) um trecho de código, que auxilie o jogador a encontrar o valor das variáveis solicitadas; iii) uma pergunta, solicitando o que o jogador deverá encontrar; e iv) uma instrução, explicando o que o jogador deve fazer para responder a esta questão. Há também um botão “Continuar”, para que seja possível fechar o quadro com a questão e respondê-la. Uma tela que mostra uma questão do jogo pode ser vista na Figura 5. O QS Game possui 21 questões, as quais auxiliam o jogador a ordenar o vetor de elementos apresentado no início do jogo (Figura 6) e a entender o funcionamento do algoritmo do método de ordenação QuickSort.

**Figura 4** - Exemplo de mensagem do jogo QS Game



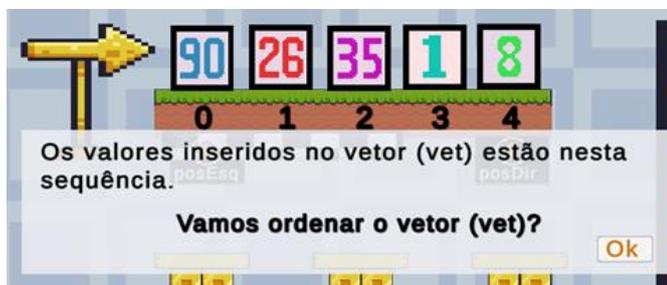
Fonte: Elaborada pelos autores

**Figura 5** - Exemplo de questão do jogo QS Game



Fonte: Elaborada pelos autores.

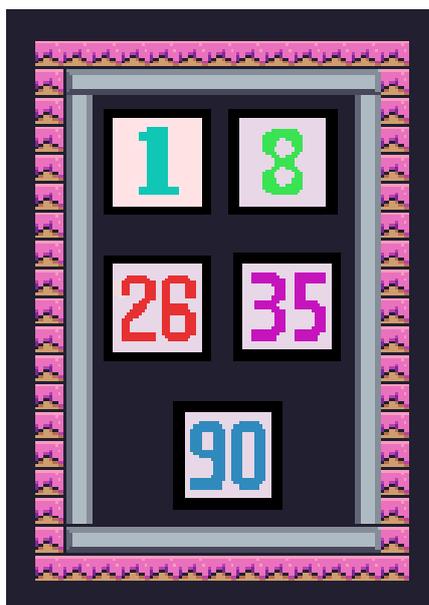
**Figura 6** - Vetor a ser ordenado no jogo QS Game



Fonte: Elaborada pelos autores.

Outro importante recurso do QS Game é o inventário, onde são armazenados os blocos com os valores a serem ordenados. Esses blocos são os elementos que o jogador deve utilizar, através da mecânica de clicar e arrastar, para definir as respostas correspondentes a cada questão. Desse modo, o usuário pode arrastar um bloco do inventário até o suporte solicitado pelo jogo (Figura 7), conforme o valor da variável solicitada pela questão apresentada.

**Figura 7** - Inventário do jogo



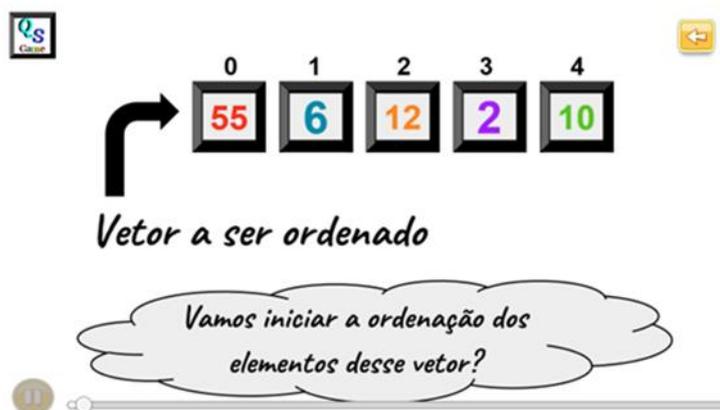
Fonte: Elaborada pelos autores.

### 3.4 Vídeo explicativo

Para facilitar o entendimento do funcionamento do algoritmo QuickSort, bem como da dinâmica do jogo, foi integrado ao QS Game um vídeo explicativo. A tela inicial do vídeo é apresentada na Figura 8. A Figura 9 mostra a tela principal do vídeo, estruturada de forma similar a interface definida para o jogo, mas utiliza um conjunto de valores diferentes.

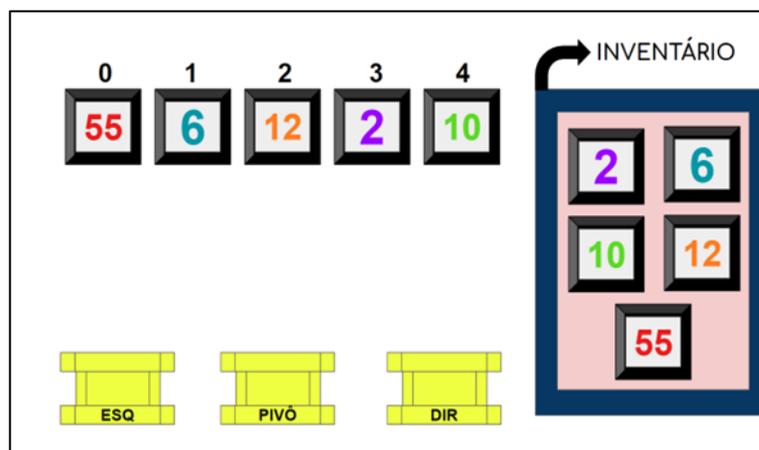
Em diversos momentos do vídeo, mensagens sobre a execução do algoritmo são apresentadas, conforme demonstrado na Figura 10. Desse modo, o vídeo pode auxiliar tanto na compreensão da lógica do algoritmo de ordenação, como no entendimento do fluxo de execução do jogo.

**Figura 8** - Tela inicial do vídeo explicativo



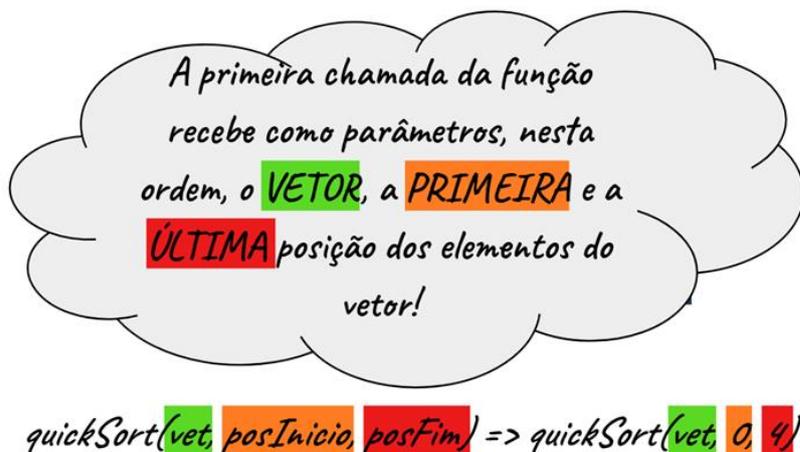
Fonte: Elaborada pelos autores.

**Figura 9** - Tela representando um momento do vídeo explicativo



Fonte: Elaborada pelos autores.

Figura 10 - Tela representando mensagens do vídeo explicativo



Fonte: Elaborada pelos autores.

### 3.5 Ferramentas de tecnologia utilizadas na implementação do jogo

O jogo foi implementado na ferramenta Unity<sup>4</sup>, um ambiente de desenvolvimento de jogos (games) e criação de animações. A Unity oferece recursos para a criação de jogos em 2D ou 3D, suporte para desenvolvimento de jogos multiplataforma, possui uma versão gratuita, além de oferecer suporte para diversas ferramentas que oferecem diversas funcionalidades para desenvolvimento de jogos digitais.

A criação dos muitos dos elementos gráficos do jogo foi realizada utilizando a ferramenta Piskel<sup>5</sup>. Piskel é um editor online e gratuito que permite a criação de *sprites* animados e *pixel arts* para serem utilizados no desenvolvimento de jogos digitais. *Pixel art* é a construção de um objeto utilizando pixels, sendo que um pixel é o menor ponto de uma imagem. *Sprites* animados são dois ou mais quadros, compostos por pixels, em sequência, possibilitando a animação de um determinado objeto.

## 4 Resultados e discussões

O QS Game possibilita aos seus jogadores uma experiência diferenciada para o aprendizado desse método de ordenação. O jogo utiliza mensagens e questões para direcionar o usuário, em uma sequência lógica que permite a aprendizagem das etapas de realização do algoritmo. Também, o jogo possui elementos gráficos e sonoros que visam tornar o ambiente mais atrativo.

<sup>4</sup> <https://unity.com/>.

<sup>5</sup> <https://www.piskelapp.com/>.

É recomendado que o jogador assista o vídeo explicativo, antes de iniciar o jogo, para se ambientar com o fluxo e dinâmica do jogo, além de aprofundar seus conhecimentos sobre o algoritmo. Vale destacar que o vídeo, separadamente, também serve como uma ferramenta de estudo.

O público-alvo das ferramentas desenvolvidas são alunos de cursos de Ciência da Computação e áreas afins. Como trabalhos futuros é possível adicionar novas fases e novos conjuntos de dados ao jogo, incrementando desse modo o nível de dificuldade e aumentando as possibilidades de valores a serem ordenados.

#### **4.1 Trabalhos relacionados**

Lawrence (2004) propõe um jogo de competição para o ensino de temas relacionados com disciplinas de estrutura de dados. Entretanto, essa abordagem tem como foco a implementação de funcionalidades em mesmo jogo, com o escopo previamente definido. Desse modo, os alunos competem para criar recursos em um jogo único. Segundo Lawrence (2004) o jogo visa fomentar uma competição amigável, visando melhorar a motivação dos alunos.

Chen (2016) propõe um jogo combinatorial baseado no QuickSort, no qual os jogadores se revezam para escolher elementos pivôs da sequência, respeitando a regra de que um movimento é inválido se não reorganizar a sequência da forma correta. Além disso, um jogador pode selecionar elementos pivôs ímpares e o outro elementos pivôs pares. Entretanto, esta abordagem não é um jogo digital, mas um jogo combinatorial baseado na teoria de jogos combinatoriais, um ramo da matemática aplicada e que estuda jogos sequenciais no formato em que os jogadores se revezam de forma definida ou se movimentam no jogo para alcançar uma determinada condição.

#### **5 Considerações Finais**

O aprendizado de diversos conceitos em áreas como a Ciência da Computação é considerado complexo pelos alunos, como é o caso de algoritmos de ordenação. Nesse caso, o principal desafio é aumentar a motivação dos alunos. O uso de jogos digitais no ensino visa aumentar o engajamento e o interesse dos alunos. Entretanto, os jogos digitais devem possuir regras bem definidas para permitir uma compreensão sólida e bem fundamentada dos temas abordados.

Um dos benefícios do uso de jogos digitais na educação, é alcançar os diferentes ritmos de aprendizado dos alunos, pois possibilita ao jogador tentar novamente até obter sucesso em cada etapa. Desse modo, com o uso de jogos sérios, voltados para o ensino, os alunos podem

assumir um papel importante para aquisição dos seus conhecimentos, o que pode melhorar a experiência do aluno em todo processo, além de permitir aplicar o que está sendo aprendido em problemas e cenários reais.

Os elementos presentes nos jogos como regras, desafios, conquistas e recompensas podem ser utilizados de forma a aumentar o engajamento e o comprometimento dos alunos com o processo de aprendizado. Outros aspectos como a interface gráfica, componentes sonoros e animações também podem ser utilizados para despertar o interesse dos alunos e, assim, facilitar o ensino.

## Referências

BIGOLIN, N. M. *et al.* Metodologias Ativas de Aprendizagem: um relato de experiência nas disciplinas de programação e estrutura de dados. **Research, Society and Development**, v. 9, no1, p. e74911648-e74911648, 2020. Disponível em: <https://doi.org/10.33448/rsd-v9i1.1648>. Acesso em: 23 jul. 2024.

CHEN, A. **The Quicksort Game**. 2016. 23f. Dissertação (Mestrado em Matemática) - KTH Royal Institute of Technology, Estocolmo, 2016. Disponível em: <https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-188170>. Acesso em: 23 jul. 2024.

DICHEVA, D.; HODGE, A. Active Learning Through Game Play in a Data Structures Course. In: **Proceedings of the 49<sup>th</sup> ACM Technical Symposium on Computer Science Education**, Baltimore, 2018, p. 834–839. Disponível em: <https://dl.acm.org/doi/pdf/10.1145/3159450.3159605>. Acesso em: 24 jul. 2024.

KAPP, K. M. **The gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education**. 1a ed. Pfeiffer & Company, 2012.

KOSA, M.; YILMAZ, M.; O'CONNOR, R.; CLARKE, P. Software engineering education and games: a systematic literature review. **Journal of Universal Computer Science**, v. 22, n. 12, p. 1558-1574, 2016. Disponível em: . Acesso em: 23 jul. 2024. Disponível em: [https://doras.dcu.ie/21648/1/jucs\\_22\\_12\\_1558\\_1574\\_kosa.pdf](https://doras.dcu.ie/21648/1/jucs_22_12_1558_1574_kosa.pdf). Acesso em: 23 jul. 2024.

LAWRENCE, R. Teaching data structures using competitive games. **IEEE Transactions on Education**, v. 47, n. 4, p. 459-466, 2004. Disponível em: <https://doi.org/10.1109/TE.2004.825053>. Acesso em: 23 jul. 2024.

LEE, J.; HAMMER, J. Gamification in education: What, how, why bother?. **Academic Exchange Quarterly**, v. 15, n. 2, p. 146-151, 2011. Disponível em: <https://dialnet.unirioja.es/servlet/articulo?codigo=3714308>. Acesso em: 23 jul. 2023.

PIVA JUNIOR, D. *et al.* **Estrutura de Dados e Técnicas de Programação**. Ed. Campus, 2014.

SOMMERVILLE, I. **Engenharia de Software**. Pearson Brasil, 2019.

TENENBAUM, A.M. *et al.* **Estruturas de Dados usando C**. Ed. Makron Books, 1995.

THOMEN, M. A. F.; AGNER, L. T. W.; SOARES, I. W. Ferramenta Pedagógica para Apoio ao Ensino de Algoritmos e Estruturas de Dados. In: **Anais do XXX EAIC – Encontro Anual de Iniciação Científica**, 2021.

VILLAS, M.V. *et al.* **Estruturas de Dados: Conceitos e Técnicas de Implementação**. Editora Campus, 1993. 298 p.