

Logical Ports: Jogo didático como ferramenta de ensino do funcionamento de portas lógicas booleanas

Pedro Elias Bertolim da Silva Lombardi¹, Nairon Neri Silva¹

¹Centro Universitário Presidente Antônio Carlos (UNIPAC)-
Rua Dr. Antonio Sena Figueiredo, nº 807 - Santa Tereza I, Barbacena - MG - Brasil

pedroeliaslombardi@yahoo.com, naironsilva@unipac.br

Resumo. *O presente estudo objetiva apresentar um jogo de computador que visa auxiliar a aprendizagem dos conceitos de portas lógicas. Desse modo, irá servir tanto como uma ferramenta introdutória, como um método de prática dos conceitos. Ademais, o software do jogo tem um caráter educacional, pois aborda, através de desafios lógicos, o funcionamento das portas lógicas AND, OR, NOT, e XOR e suas interações entre si gerando as soluções (S) esperadas por cada desafio.*

Palavras Chaves: *Portas Lógicas, Jogo Educacional, Ferramenta de Aprendizagem.*

Abstract. *The present objective study presents a computer software that aims to help the learning of the concepts of logic gates. In this way, it will serve both as an introductory tool and as a method of practicing the concepts. In addition, the game software has an educational character because it addresses, through logical challenges, the operation of the logic gates AND, OR, NOT, and XOR and their intentions among themselves, generating the solutions (S) expected for each challenge.*

Keywords: *Logic Gates, Educational Game, Learning Tool.*

1. Introdução

A princípio, no trabalho de [Pessini 2018], foi realizado um estudo utilizando jogos eletrônicos dos gêneros de ação, esportes e treinamento cerebral, onde foi constatado que o uso dos mesmos tem se mostrado importante na melhora das habilidades cognitivas e auxiliado na reabilitação de funções motoras. Ou seja, os indivíduos que realizam a prática frequente de jogos eletrônicos atingem um maior estímulo cerebral e um melhor desenvolvimento de funções nervosas e motoras em comparação a indivíduos que não fazem o uso de jogos.

Segundo [Rizzo 2001], "o jogo motiva e por isso é um instrumento muito poderoso na estimulação da construção de esquemas de raciocínio, através de sua ativação. O desafio por ele proporcionado mobiliza o indivíduo na busca de soluções ou de formas de adaptação a situações problemáticas e, gradativamente, o conduz ao esforço voluntário."(p.40). Tendo isso em vista, conclui-se que o jogo eletrônico é uma ferramenta valiosa, por beneficiar em diversos aspectos a questão motora do jogador.

Tomando como referência o estudo de [da Silva and Yepes. 2018], pode-se definir o jogo como uma atividade em que o indivíduo segue regras e orientações disponibilizadas

para se alcançar o objetivo desejado, isto é, vencer. Ademais, permite que os jogadores possam explorar diferentes áreas do conhecimento como a probabilidade, a estratégia e o raciocínio lógico, contribuindo, para que uma ampla possibilidade de jogos possa ser desenvolvida.

De acordo com [Jurgina 2020], o desinteresse de alunos da área da computação na disciplina de circuitos digitais pode estar associado à necessidade de possuir conhecimentos específicos abordados em disciplinas anteriores. Portanto, é preciso estimular os alunos, de forma lúdica, a transformarem as dificuldades enfrentadas em novos conhecimentos.

Segundo [Rizzo 2001], “a atividade lúdica pode ser um eficiente recurso aliado ao educador à medida que este, esteja interessado no desenvolvimento da inteligência de seus alunos.” (p.40).

A Resolução nº 5, de 16 de novembro de 2016 [Brasil 2016], que instituiu as diretrizes curriculares nacionais (DCNs) para os cursos de graduação na área da computação no Brasil, busca padronizar e orientar a formação dos profissionais da área, considerando o avanço constante das tecnologias e a demanda crescente por especialistas qualificados. O que se espera de um formando na área da computação é o desenvolvimento da capacidade de “planejar, especificar, projetar, implementar, testar, verificar e validar sistemas de computação (sistemas digitais)” [Brasil 2016], isto é, ao decorrer da graduação os alunos irão se deparar com disciplinas que abordam os princípios da álgebra booleana e suas aplicações em circuitos digitais.

O presente trabalho dispõe-se a apresentar o “Logical Ports”, um jogo digital de caráter educacional desenvolvido com o intuito de servir como uma porta de entrada para os conceitos e funcionamento das portas lógicas booleanas. O jogo utiliza essas portas em desafios de *puzzle* (quebra-cabeças), com foco na criação de circuitos combinacionais capazes de executar expressões booleanas propostas em cada nível. Dessa forma, o objetivo é tornar o aprendizado de circuitos combinacionais mais acessível e interessante para estudantes e entusiastas do assunto.

2. Estado da arte

Durante o processo de busca por trabalhos relacionados, foram encontradas algumas pesquisas e programas que se propõem a simular o funcionamento das portas lógicas, havendo também alguns *softwares* de jogos. A seguir serão apresentados alguns dos trabalhos encontrados.

Em [Pereira 2006] foi realizado um estudo que visava demonstrar a importância da prática dos conceitos de circuitos digitais. Ademais, o trabalho evidencia a necessidade de se equilibrar os aspectos teóricos e práticos durante o processo de aprendizagem, constataando que muitos dos egressos apresentavam dificuldades nas aulas teóricas, onde alguns só compreendiam os conceitos dados em aulas práticas com exemplos de aplicações. Portanto, os autores concluem que com o uso de métodos práticos e ferramentas no ensino dos sistemas digitais o interesse pelos formandos aumentou, assim como a prática através de exercícios contribuíram para uma maior fixação dos conceitos.

O trabalho apresentado por [Jurgina 2020] tem o intuito de demonstrar a ferramenta *LogicFlow* e sua aplicação no ambiente de aprendizagem. Assim sendo, o *Logic-*

Flow trata-se de um *software* voltado para dispositivos móveis de interface simplificada, que permite ao usuário simular circuitos digitais para o desenvolvimento de seus conhecimentos. Por fim, o trabalho conclui que o uso de ferramentas semelhantes no ambiente de sala de aula, pode auxiliar professores a aumentar o engajamento dos alunos nos assuntos associados às portas lógicas.

Em [Becker 2014], foi realizada uma pesquisa na qual foram avaliadas diferentes ferramentas de simplificação de circuitos lógicos combinacionais, objetivando elencar *softwares* com potencial educacional no curso de Engenharia de Computação. Após a avaliação de critérios como usabilidade, metodologia e layout da aplicação, foram realizados testes práticos, tanto presenciais em laboratório quanto à distância em aulas *online*. Além disso, também foi utilizado um questionário respondido pelos alunos do curso, o qual permitiu constatar que o uso das ferramentas avaliadas no trabalho aumentou o interesse e a curiosidade, além de contribuir de forma positiva no aprendizado da álgebra booleana e dos circuitos.

Em [Miquelini and Ferrari. 2021] é descrito a existência de dificuldades na aprendizagem de lógicas de circuitos digitais devido à complexidade da teoria, e ressalta a importância da prática e como os simuladores permitem um maior acesso à mesma. Tendo isso em vista, o trabalho apresenta o LOGISIM, destacando sua acessibilidade, por ser um *software* gratuito, e a necessidade de ferramentas como essa no ramo acadêmico, pois ela distancia o pensamento de que o ensino destas tecnologias está diretamente ligado às universidades. Logo, o trabalho demonstra que LOGISIM é uma ferramenta com diversas versões e bibliotecas que visam atender aos mais diversos objetivos, e conclui que apesar das limitações que o *software* possui, ele cumpre seu propósito como um simulador lógico.

Como apontado nos trabalhos acima, o uso de ferramentas de prática de conteúdos teóricos possibilitou um melhor aproveitamento e desenvolvimento da aprendizagem dos usuários. Utilizando o princípio da eficácia desse método, o presente trabalho diferencia-se dos apresentados anteriormente, pois possui seu foco em desenvolver o conhecimento dos usuários de forma gradativa, através de níveis de dificuldade focados na criação de circuitos combinacionais capazes de executar expressões booleanas predeterminadas em cada desafio.

3. Materiais e métodos

Para o desenvolvimento do *software* foi pesquisado o funcionamento dos circuitos lógicos da álgebra booleana, e com base nisso, foram criados diversos desafios, separados em diferentes níveis no jogo. Para a programação do jogo foi utilizado o motor de desenvolvimento Godot 3.5 com a escolha de linguagem de programação sendo o GDScript e também a ferramenta Aseprite para o design dos componentes do projeto. Esta seção é voltada a apresentar estas ferramentas e a importância das mesmas no desenvolvimento deste trabalho.

3.1. Álgebra Booleana

Álgebra booleana é um sistema que busca representar regras através de símbolos, codificando proposições, ou seja, uma solução esperada que possa ser verificada como falso ou verdadeiro, 0 e 1 respectivamente.

As operações lógicas básicas são conhecidas como operadores booleanos que são: AND, NOT e OR (e, não e ou) [Vieira 2000], estas operações podem ser representadas através de símbolos, sendo esses conhecidos como portas lógicas, que segundo [Aguiar 2017] "são componentes fundamentais na criação destes circuitos digitais".

3.1.1. Porta NOT (porta não ou inversor)

A porta NOT inverte o valor da variável de entrada. Caso a entrada seja verdadeira, sua saída será falsa; caso sua entrada seja falsa, a saída será verdadeira. Como pode ser observado na Figura 1, no caso de uma entrada representada pela variável A a saída será A' .



Figura 1. Porta NOT. Fonte: Elaborada pelo autor

3.1.2. Porta AND (Porta E)

A porta AND produz uma saída verdadeira, apenas quando todos os valores de entrada forem verdadeiros. É representada pelo sinal matemático de multiplicação (*). Logo, conforme a Figura 2 se os valores de entrada forem A e B, o valor da saída será $(A*B)$.

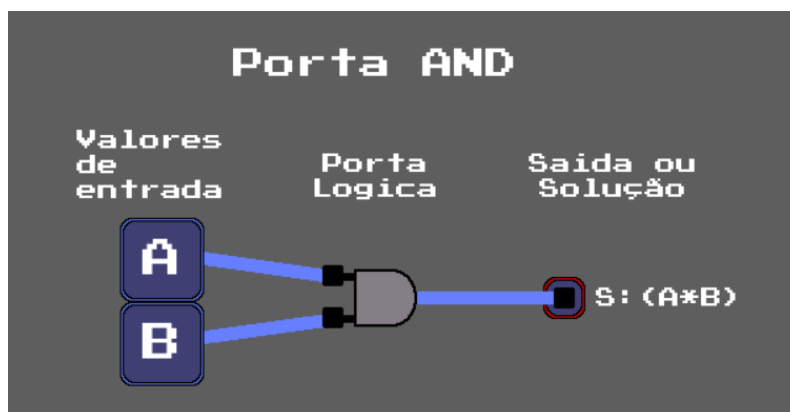


Figura 2. Porta AND. Fonte: Elaborada pelo autor

3.1.3. Porta OR (Porta OU)

Realiza a operação "OU", caso algum dos valores da entrada seja verdadeiro, a saída será verdadeira; caso ambos os valores de entrada sejam falsos, a saída será falsa. É representada pelo sinal matemático de adição (+). Conforme pode ser observado na Figura 3, se os valores de entrada forem A e B a saída será $(A+B)$.

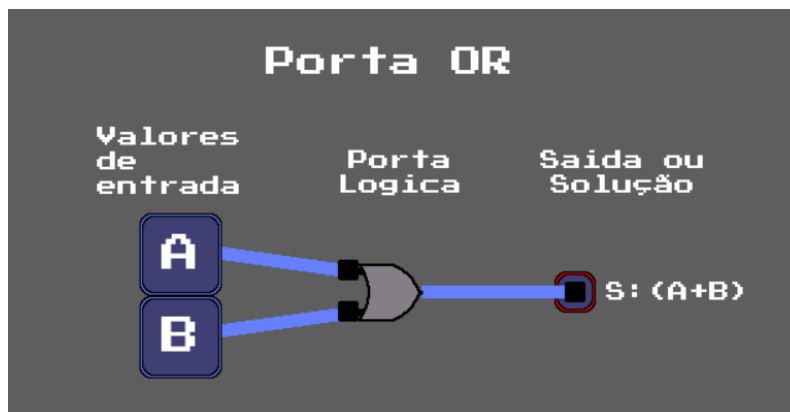


Figura 3. Porta OR. Fonte: Elaborada pelo autor

3.1.4. Porta XOR (Porta OU-exclusivo)

A porta XOR (ou OU-exclusivo) possui a função de produzir uma saída verdadeira quando as entradas forem diferentes; Se as entradas forem iguais, a saída será falsa. É representada pelo sinal matemático de soma direta (\oplus). Como exemplificado na Figura 4, se as entradas forem A e B, a saída será $(A \oplus B)$.

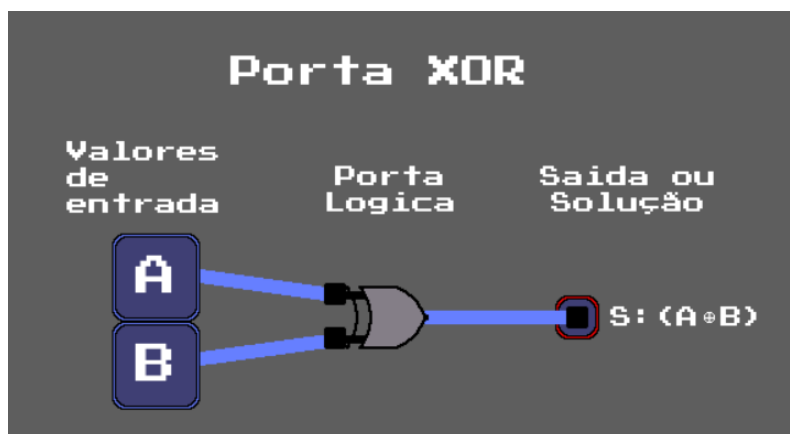


Figura 4. Porta XOR. Fonte: Elaborada pelo autor

Através do uso das portas lógicas é possível criar circuitos combinacionais que resultam em expressões booleanas, que segundo [Floyd 2009] são estruturas lógicas capazes de assumir dois valores, sendo eles verdadeiro ou falso. Ou seja, com uma expressão booleana é possível criar o circuito lógico que a representa [Floyd 2009]. Portanto, o

foco dos desafios apresentados no Logical Ports é a criação dos circuitos combinacionais capazes de representar as expressões booleanas propostas.

3.2. Godot

Segundo [Furtado and Santos 2002], um motor de desenvolvimento de jogos é uma ferramenta capaz de oferecer ao programador um conjunto de funcionalidades, que permite ao mesmo, especificar em alto nível, como cada elemento deve se comportar no jogo, sem exigir conhecimento prévio de como esses elementos são implementados.

Ademais, o Godot é um motor de desenvolvimento gratuito, de código aberto, voltado tanto para o desenvolvimento 2D(bidimensionais) quanto 3D (tridimensionais). Sua premissa é ser uma ferramenta com aprendizagem simplificada que permita o desenvolvimento de jogos eletrônicos para múltiplas plataformas sem restrições. A linguagem utilizada neste trabalho foi a GDScript, que é a linguagem padrão da plataforma, porém a mesma possui suporte às linguagens Visual Scripting, C# e C++ [Linietsky 2022].

Os principais fatores que levaram o Godot a ser escolhido como ferramenta para o desenvolvimento do Logical Ports, foram a interface intuitiva e de fácil aprendizagem, e o bom desempenho que a ferramenta apresenta na criação de jogos 2D.

3.3. GDScript

O GDScript é uma linguagem de alto nível, com sintaxe similar a Python em sua identificação. Portanto, seu principal objetivo é ser uma estrutura otimizada integrada ao Godot Engine, permitindo maior flexibilidade no desenvolvimento de conteúdo [Linietsky 2022].

Em [da Silva and Yepes. 2018] é afirmado que foram testadas diferentes linguagens de terceiros, dentre elas as linguagens Lua e Python, porém, concluiu-se que a criação de uma linguagem própria permitiria um desempenho superior, fator que levou a mesma a ser escolhida para este projeto. Deste modo, o GDScript foi incorporado como linguagem primária do Godot. Todavia, apesar da ferramenta possuir suporte a outras linguagens, a mesma é fortemente associada a sua linguagem padrão [Linietsky 2022].

4. Resultados

Após a realização do projeto, cabe-se nessa seção abordar os resultados do desenvolvimento do *software*.

4.1. A aplicação

O funcionamento da ferramenta Logical Ports se dá ao iniciar o programa, onde o usuário é apresentado à tela inicial (Figura 5), contendo as opções: Jogar, que leva aos desafios; Sobre as portas, que é a sessão destinada a explicar o funcionamento das portas lógicas; Como jogar, onde o usuário poderá se aprofundar sobre o funcionamento das mecânicas e funcionalidades do jogo; Sair, onde o jogador pode fechar o programa.

Vale ressaltar que alguns fatores considerados durante a concepção inicial do projeto não foram implementados devido a problemas no desenvolvimento. Por exemplo, a marcação de níveis concluídos e a abordagem dos desafios relacionados à simplificação de expressões booleanas não foram incluídas. No entanto, esses fatores podem vir a ser trabalhados em atualizações futuras.



Figura 5. Tela inicial. Fonte: Elaborada pelo autor

4.1.1. Jogar

Ao selecionar a opção jogar, o usuário será levado a uma tela de seleção de fases (Figura 6) que possui a opção de retornar ao menu inicial e também dá acesso a cada nível do jogo. Como o objetivo do jogo é permitir a prática da criação de circuitos combinacionais, o usuário já terá por padrão acesso a todos os níveis disponíveis, podendo assim se desafiar na complexidade que julgar mais adequada a seu conhecimento.

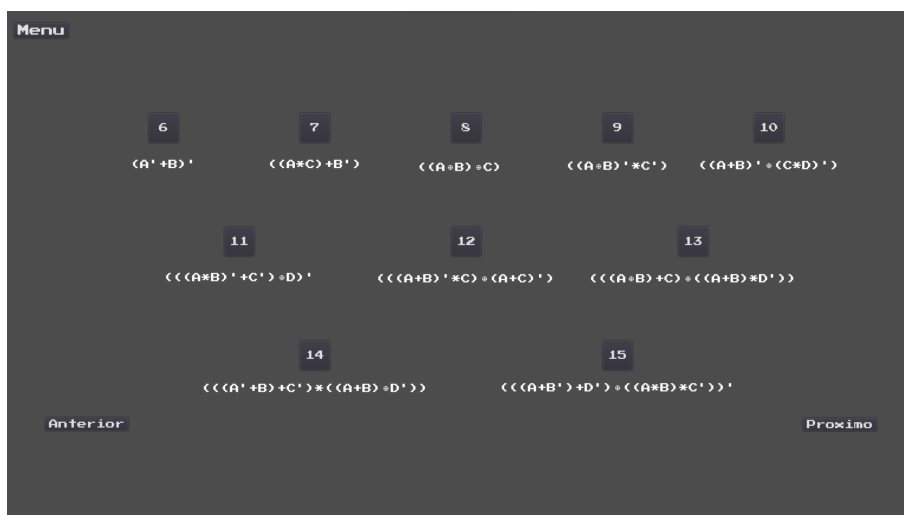


Figura 6. Seleção de fases. Fonte: Elaborada pelo autor

4.1.2. Desafios

Ao escolher um nível, o usuário deverá utilizar as portas lógicas disponíveis em cada nível para montar um circuito combinacional capaz de executar as expressões booleanas propostas (S). A quantidade de componentes a ser utilizado pelo jogador é limitado, podendo

ser iguais ou maior que a quantidade necessária para solucionar o desafio. Ao montar um circuito combinacional válido, o indicador da solução esperada, no centro superior da tela, ficará com a coloração verde e aparecerá, no canto inferior direito, as opções para voltar a seleção de fases, ir pra tela inicial ou prosseguir para o próximo desafio. Caso o circuito montado pelo jogador não seja capaz de executar a expressão booleana esperada, o indicador da mesma será alterado para a cor vermelha, e aparecerá logo acima do valor de saída a mensagem "Parece que algo não está certo, verifique com atenção" também na cor vermelha com o intuito de sinalizando ao jogador a necessidade de realizar as devidas alterações para que o nível seja concluído.

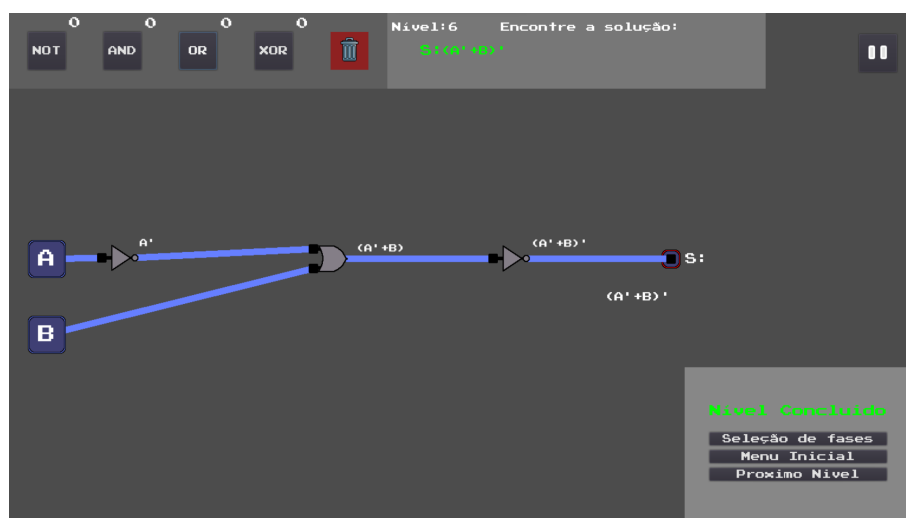


Figura 7. Exemplo de nível. Fonte: Elaborada pelo autor

A Figura 7 demonstra um dos níveis do jogo, onde o jogador deve montar o circuito combinacional correspondente à expressão booleana proposta, ou seja $S:(A' + B)'$. Para criar o circuitos correspondente, o jogador pode-se utilizar dos botões NOT, AND, OR e XOR para gerar as portas que deseja utilizar, sendo essas limitadas pelos números que aparecem no canto superior direito de cada botão. Ao gerar uma porta, o contador respectivo a ela é subtraído. Caso o número seja zero, o jogador não poderá gerar mais componentes referentes a ela, a menos que delete uma do mesmo tipo, arrastando-a até a lixeira. Para organizar e gerar o circuito de forma mais visível, o usuário deverá mover, com o mouse os componentes que decidir utilizar e os conectar usando os cabos azuis de cada entrada e saída.

4.1.3. Sobre as portas

Ao selecionar a opção "Sobre as portas" na tela inicial, o usuário será direcionado a uma seção onde poderá consultar informações a respeito do funcionamento de cada porta lógica de forma individual (figura 8). Entretanto, no ambiente de jogo, as portas podem ser utilizadas em sequência, podendo assim representar novos resultados. Por exemplo, é possível utilizar as portas NOT e AND para obter um resultado NAND. Contudo, essas portas (NAND, NXOR e NOR) não possuem representações ilustradas dentro do jogo.

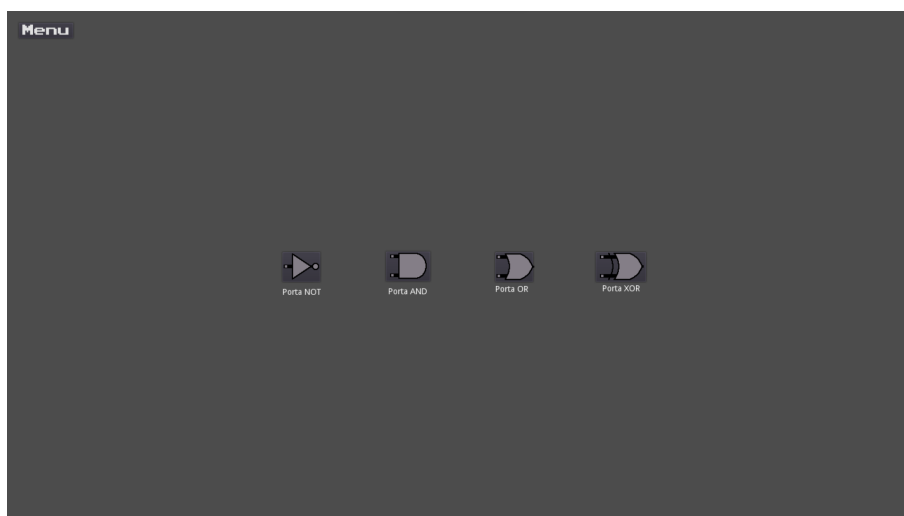


Figura 8. Sobre as portas. Fonte: Elaborada pelo autor

5. Conclusão

Levando em consideração o conteúdo apresentado, atividades lúdicas são ferramentas eficientes para o aprendizado, bem como a utilização de metodologias práticas para reforçar conhecimentos teóricos. Tendo em vista a forte capacidade de jogos eletrônicos em gerar interesse em determinados assuntos por meio de desafios, o presente trabalho buscou desenvolver um jogo capaz de apresentar os conceitos das portas lógicas e seu uso na criação de sistemas combinacionais, de uma forma acessível e atrativa.

Ao fim do desenvolvimento do projeto, foi possível constatar que o objetivo inicial foi alcançado, pois o jogo oferece um conjunto de informações e desafios práticos que permitem ao usuário tanto se familiarizar com os conceitos de circuitos combinacionais com portas lógicas, quanto praticar esses conhecimentos.

No decorrer do desenvolvimento do *software*, foi possível observar a importância da experimentação de diferentes métodos e ferramentas na criação do programa. Isso se deve ao fato de que, apesar de o Godot possuir uma documentação oficial robusta e muitos conteúdos disponibilizados pela comunidade, os mesmos não abrangem todas as possibilidades que a *engine* utilizada possibilita. Assim, a busca por alcançar resultados satisfatórios, permitiu adquirir uma maior capacidade criativa na solução de problemas, uma vez que foi necessário adaptar as funcionalidades disponíveis para executar novas funções capazes de atender aos resultados desejados.

É importante salientar que o jogo possui potencial para diversas aplicações futuras voltadas à acessibilidade e ao ensino. Seria possível considerar a adaptação do jogo para aplicações *web* e dispositivos móveis, aproveitando as capacidades do motor de desenvolvimento utilizado. Além disso, seria interessante a implementação de mais conceitos abrangidos pela álgebra booleana, como a simplificação de circuitos, a inclusão de mais portas existentes, tratamento de tabelas verdade e explicações sobre o seu funcionamento. Para tornar a ferramenta mais completa, também poderia ser elaborado um sistema que permitisse ao jogador salvar seu progresso, medir seu desempenho e publicar seus resultados, para que essas informações possam ser utilizadas para melhorar o programa em versões futuras.

Referências

- Aguiar, A. F. (2017). Estudo de propagação de plásmons de superfície em grafeno e realização de portas lógicas or, and e xor. DETE - Dissertações defendidas na UFC.
- Becker, L. e. a. (2014). Um estudo sobre a adoção de softwares para simplificação de expressões booleanas, como objetos de aprendizagem, em um curso de engenharia de computação.
- Brasil (2016). Resolução nº5, de 16 de novembro de 2016, institui as diretrizes curriculares nacionais para os cursos de graduação na área da computação. Brasil. Diário Oficial da União. Brasília, DF, n. 220, 17 novembro 2016. Seção I, p.22.
- da Silva, V. P. and Yepes., I. (2018). Desenvolvimento de jogos na plataforma godot. IX EATI–Encontro Anual de Tecnologia da Informação: 102.
- Floyd, T. (2009). Sistemas digitais: fundamentos e aplicações. Bookman Editora.
- Furtado, A. W. B. and Santos, A. L. M. (2002). Fungen–um motor para jogos em haskell. dimensions 2: 2.
- Jurgina, Laura Quevedo, e. a. (2020). logicflow: Uma ferramenta para o auxílio de ensino-aprendizagem de circuitos digitais. Anais do XXXI Simpósio Brasileiro de Informática na Educação. SBC.
- Linietsky, Juan; Manzur, A. (2022). Godot engine last documentation. Disponível em: https://docs.godotengine.org/pt_BR/latest/, Consultado em: 08 nov 2022.
- Miquelini, R. a. A. and Ferrari., H. O. (2021). Logisim: Ferramenta para simulação de circuitos combinacionais e sequenciais digitais. Intercursos Revista Científica 20.2.
- Pereira, Mauro Conti, e. a. (2006). *Motivando Alunos em Circuitos Digitais: um Estudo de Caso*. Anais do Workshop sobre Educação em Computação.
- Pessini, Rodrigo Antonio, e. a. (2018). Análise da plasticidade neuronal com o uso de jogos eletrônicos. Journal of Health Informatics 10.1.
- Rizzo, G. (2001). Jogos inteligentes: a construção do raciocínio na escola natural. Bertrand Brasil.
- Vieira, F. M. S. (2000). Álgebra booleana. Educação & Tecnologia 5.1.