

AI(3P)A: Uma Metodologia para o Ensino de Lógica de Programação Utilizando Jogos Eletrônicos

Leonardo M Faêda

Departamento de Informática
Universidade Federal de Viçosa (UFV)
Viçosa, MG, Brasil
leonardo.faeda@gmail.com

Matheus F O Baffa

Dep. de Computação e Matemática
Universidade de São Paulo (USP)
Ribeirão Preto, SP, Brasil
mfreitas826@gmail.com

Julie S Pereira

Dep. Acadêmico da Ciência da Computação
IF Sudeste MG
Rio Pomba, MG, Brasil
juliespereira7@gmail.com

Abstract—This paper presents the teaching method AI(3P)A. This method combines Playful Class (A), Idealization (I), Planning (P), Assisted Practice (P), Individual Practice (P) and Application (A). The goal of this methodology is to help teachers in his pedagogical activities, guiding on how to present new content, plan and divide tasks, practices and assessments, using electronic game's development tools. Two different contexts for evaluating the method are presented in this work: one composed of students aged between eight and twelve years old and the other aged between thirteen and sixteen years old. The analysis of the proposed method was carried out through self-assessment questionnaires and activities evaluations. In both classes the method was well accepted by students in satisfaction, motivation and control, satisfying the expectation in learning the concepts of programming.

Index Terms—games in education, programming logic, teaching methodology

Resumo—Este trabalho apresenta o método de ensino AI(3P)A. Este método congrega Aula Lúdica (A), Idealização (I), Planejamento (P), Prática Auxiliada (P), Prática Individual (P) e Aplicação (A). O objetivo desta metodologia é auxiliar o professor em suas atividades pedagógicas, orientando-o a como apresentar novos conteúdos, planejar e dividir suas tarefas, práticas e avaliações, utilizando ferramentas de desenvolvimento de jogos eletrônicos. Dois contextos diferentes de avaliação do método são apresentados neste trabalho: um composto por alunos de faixa etária de oito a doze anos e outro na faixa etária de treze a dezesseis anos. A análise do método proposto foi realizada através de questionários de autoavaliação e avaliações das atividades desenvolvidas. Em ambas as turmas o método foi bem aceito pelos alunos em satisfação, motivação e controle, satisfazendo a expectativa no aprendizado dos conceitos de programação.

Palavras Chave—jogos na educação, lógica de programação, metodologia de ensino

I. INTRODUÇÃO

As Tecnologias da Informação e Comunicação (TIC) estão cada vez mais presentes entre as crianças, que nascem imersas no mundo digital, aparentando ser fluentes com as novas tecnologias [6]. No entanto, Mota et al. [5] alega que para possuir o domínio sobre as novas tecnologias, não basta saber manipular-las, é necessário dominar a projeção e criação de novos programas e jogos, tarefa a qual necessita de conhecimento em lógica de programação.

O conhecimento de programação aperfeiçoa as habilidades relacionadas ao raciocínio lógico e a matemática, importantes para muitos outros aprendizados, além do desenvolvimento do pensamento computacional [7]. Este, é um pensamento característico de pessoas da área de programação, mas cabível para qualquer pessoa e abrange um conjunto de elementos [11].

O pensamento computacional descrito por Wing [11] envolve a resolução de problemas aparentemente difíceis, eficientemente, reformulando-o em um problema que sabemos resolver, utilizando a redução, incorporação, transformação ou simulação. A importância de estimular o pensamento computacional está em utilizar as habilidades de criação de programas computacionais como metodologia para resolver problemas mais complexos. Um exemplo na computação é o *hashing*. O *hashing* é um método que permite organizar uma grande quantidade de dados, onde o conjunto de dados é dividido em subconjuntos de mais fácil acesso. O método *hashing* utilizado por crianças para organizar seus blocos de peças de lego, influencia no tempo para encontrar uma peça correta [12].

Os principais elementos que fundamentam o pensamento computacional são (i) a decomposição; (ii) o reconhecimento de padrões; (iii) a abstração e; (iv) algoritmos. A decomposição consiste em dividir um problema completo em subproblemas mais fáceis de serem gerenciados. O reconhecimento de padrões consiste em assimilar problemas que já foram solucionados com o problema que está sendo abordado. A abstração consiste em reduzir a concentração nos dados mais relevantes e algoritmos é a solução para a resolução do problema. Esses elementos foram associados em cada etapa do método de ensino proposto neste trabalho.

Essas características mostram a relevância da inclusão do ensino de lógica de programação como parte do ensino básico dos alunos, desde as primeiras etapas do seu processo de aprendizado. E, para que essa prática seja capaz de acontecer de forma efetiva e divertida, é indispensável a utilização de ferramentas que possam auxiliar professores neste processo.

Neste trabalho apresentamos o AI(3P)A, um método de

ensino que auxilia o professor em suas atividades pedagógicas, ajudando-o em como apresentar novos conteúdos, dividir e planejar tarefas, práticas e avaliações, fazendo uso de ferramentas de desenvolvimento de jogos eletrônicos. O método possui o objetivo de proporcionar ao aluno o primeiro contato com os conceitos de programação e lógica por uma abordagem lúdica. O método foi aplicado a dois ambientes diferentes.

Este trabalho está organizado da seguinte forma: na Seção 2, faremos uma revisão bibliográfica da utilização de ferramentas de jogos eletrônicos e métodos de ensino de lógica de programação; na Seção 3, apresentamos a sequência didática do método de ensino AI(3P)A; na Seção 4 é apresentado o plano de estudos utilizadas nas aulas; na Seção 5, relatamos a experiência utilizando o método proposto e por fim; na Seção 6, apresentamos nossas conclusões e sugerimos novas investigações.

II. TRABALHOS CORRELATOS

Vários estudos foram realizados com o objetivo de estabelecer métodos de ensino e práticas pedagógicas capazes de educar de maneira lúdica. Estas propostas utilizam diversas ferramentas no auxílio ao aprendizado de conceitos de computação e lógica de programação.

Em Santos et al. [8] é apresentado um método de ensino denominado de EDATM. Esse método possui o objetivo de auxiliar professores em atividade pedagógicas no ensino de lógica de programação utilizando robótica educacional. Dois contextos diferentes de avaliação do método são apresentados no trabalho: alunos do ensino fundamental e ingressantes do ensino superior. Os resultados foram coletados a partir de autoavaliação dos sentimentos dos alunos, satisfazendo a expectativa no aprendizado de conceitos de programação.

No trabalho de Viana e Portela [10] é abordado o uso de *softwares* educativos para apoiar a introdução à lógica e algoritmos, tanto no ensino de base quanto no superior. Entre os *software* utilizados está o *Visualg*, um programa que edita, interpreta e executa algoritmos escritos em pseudocódigo (Portugol). Os resultados desta análise apontaram que alguns dos *softwares* utilizados são mais adequados para o ensino fundamental e médio e outros para a graduação. Por fim, ele sugere propostas de abordagens para adoção dessas ferramentas em sala de aula.

O estudo de Casarotto et al. [2] apresenta uma proposta de jogo de tabuleiro educacional, denominado de *Logirunner*, que visa auxiliar o aprendizado de algoritmos e lógica de programação. Um protótipo do jogo *Logirunner* foi idealizado com elementos físicos reais, cartas, peças e tabuleiro, para ser utilizado em sala de aula. Foram inseridos em suas regras, elementos de lógica de programação e algoritmos além de conteúdos a serem reforçados durante as sessões de jogos. Nos resultados obtidos, os alunos consideraram a interação, diversão e aprendizagem como os pontos fortes do jogo.

Os estudos relatados acima apresentam algumas maneiras de introduzir conceitos de programação utilizando vários tipos de ferramentas como robótica, jogos não-digitais e *softwares*. Neste trabalho, o método proposto foi aplicado em dois

contextos diferentes e utilizou ferramentas de desenvolvimento de jogos eletrônicos para auxiliar no ensino de lógica de programação. Diversos trabalhos abordaram essas ferramentas na educação de programação.

Dantas et al. [3] relata a experiência de licenciandos em Computação em ensinar lógica de programação no ensino fundamental. Nessa prática foi utilizado um jogo educativo digital, denominado *Robotizen*, visando constituir um ambiente de aprendizado lúdico. O método de ensino aplicado baseou-se pelas fases de desenvolvimento cognitivo das crianças e na divisão estratégica dos cenários do *Robotizen*, assim introduzindo conceitos de programação de forma divertida. Os resultados apontaram um potencial do uso do jogo para engajar e estabelecer aprendizagem orientada a interesses e descobertas.

O estudo de Ferreira et al. [4] possui o objetivo de despertar raciocínio lógico de jovens e adultos das comunidades da cidade de Salvador, sem conhecimento prévio de programação utilizando a plataforma de desenvolvimento de jogos *Scratch*. Também foi utilizado o método de ensino proposto por [1], que tem como objetivo ensinar computação sem o uso do computadores ou quaisquer recursos tecnológicos. Os resultados apresentados mostraram que o método proposto pelos autores despertou um grande interesse dos alunos em programação.

Uma revisão sistemática da literatura foi realizada por Souza e Castro [9] sobre metodologias que utilizam a ferramenta *Scratch* no ensino de programação para crianças. O trabalho abordou estudos dos anos de 2007 até 2016. Na revisão foi possível notar que estudos brasileiros possuem dificuldade de disseminar a cultura do ensino de programação, sendo a baixa infraestrutura e a falta de treinamento para professores os principais fatores. Esta revisão mostra que metodologias para o ensino de programação aliados à mídias interativas e lúdicas, apresentam resultados positivos quando aplicadas no contexto prático.

III. A SEQUÊNCIA DIDÁTICA PROPOSTA

Nesta seção apresentamos o método de ensino AI(3P)A uma abreviação das etapas deste modelo que significam: Aula Lúdica, Idealização, Planejamento, Prática Auxiliada, Prática Individual e Aplicação. O AI(3P)A é um método de ensino aplicado para o ensino de lógica de programação utilizando ferramentas de jogos eletrônicos. As ferramentas de desenvolvimento de jogos utilizadas nessa metodologia são o *Scratch* e o *Construct*, ambas as ferramentas foram escolhidas devido sua facilidade em realizar a programação dos jogos através de blocos de códigos. A escolha de utilizar duas ferramentas é devido a limitação da ferramenta *Scratch* em questão da utilização de funções e componentes.

A Fig. 1 apresenta uma visão geral de como são aplicadas as fases do AI(3P)A. A primeira fase é a aula lúdica seguida pela idealização do jogo proposto, utilizando um determinado planejamento. A fase seguinte é então a prática auxiliada pelo professor para desenvolver o jogo proposto. As etapas de idealização e planejamentos são repetidas para a realização da prática individual do jogo. Por fim, é realizado a etapa de

aplicação que se entende como uma dinâmica a qual os alunos testam os jogos desenvolvidos por eles.

A. Aula Lúdica

A aula lúdica possui o objetivo de inserir um novo conteúdo de programação. Para que a aula não seja realizada apenas na forma de exposição narrativa, a qual o professor apresenta o conteúdo sem uma dinâmica com os alunos, o conceito apresentado é relacionado e discutido com exemplos do cotidiano.

Dessa maneira, a aula acontece de uma forma dialogada, possuindo uma contribuição direta do professor e do aluno na apresentação e absorção do novo conteúdo. Assim, o conteúdo antes visto como complicado, passa a ser mais fácil de ser absorvido pois é apresentado aos alunos de uma maneira lúdica e com exemplos táteis.

Essa etapa esta diretamente relacionada com o elemento de abstração do pensamento computacional, proposto por Wing [11]. De forma análoga, este elemento propõe apresentar o conceito sem focar em seus detalhes.

B. Idealização

Na etapa de idealização, um jogo é proposto pelo professor para ser desenvolvido juntamente com os alunos nas etapas seguintes. O objetivo dessa etapa é retirar as ideias principais do jogo e relacionar o conteúdo apresentado na etapa anterior. Desta forma é realizada uma aplicação direta do novo conceito com a proposta. Note que nesta etapa, o professor deve instigar o pensamento dos alunos e relacionar o conceito com o jogo desenvolvido.

A relação da etapa de idealização com os elementos do pensamento computacional [11] é novamente a abstração pois o jogo é planejado sem aprofundar em detalhes. Essa etapa tem um papel fundamental na integração do conteúdo apresentado em um contexto do cotidiano com o ambiente do jogo proposto.

C. Planejamento

Na etapa de planejamento, o objetivo é realizar o detalhamento de como o jogo será desenvolvido. Aqui, são abordados quais são os cenários, personagens, ações e objetivos. As atividades são divididas em fases de desenvolvimentos. A relação entre as atividades são monitoradas de forma que elas não sejam desenvolvidas repetitivamente.

Esta etapa esta diretamente relacionada com dois elementos do pensamento computacional: decomposição, no sentido de planejar as atividades em pequenas partes e as acoplado de forma gradativa, e reconhecimento de padrões que ações que são utilizadas mais de uma vez, podem ser reaproveitadas.

D. Prática Auxiliada

A etapa de prática auxiliada possui o objetivo de desenvolver o jogo proposto pelo professor. Nela os alunos são orientados no desenvolvimento da aplicação utilizando os novos conceitos aprendidos. A prática proporciona ao aluno, raciocínio lógico de programação o qual ele poderá elaborar suas regras de construção do jogo. Desta maneira, os alunos

são preparados para a realização da próxima etapa, na qual devem propor e desenvolver um jogo individualmente.

Os elementos do pensamento computacional abordados nessa etapa são (i) decomposição, em razão de que o aluno se organiza em passos menores e gradativos durante o desenvolvimento da aplicação proposta, e (ii) algoritmo que é a estrutura base para a solução para o desenvolvimento das ações presentes no jogo, o qual há uma ordem nos passos antes da ação a ser executada.

E. Prática Individual

A etapa da prática individual possui um caráter avaliativo. Nela os alunos devem compartilhar ideias e propor um novo jogo utilizando os conceitos aprendidos. Após a escolha do jogo que será desenvolvido, as fases de idealização e planejamento são realizadas novamente para a prática individual. Ao termino dessas etapas, os alunos começam a implementar o jogo individualmente, colocando em prática os conhecimentos adquiridos.

A missão do professor nessa etapa é intermediar na escolha da proposta de jogo, observando a viabilidade e se a proposta se encaixa no conteúdo da aula e no tempo disponível para a realizar a prática. O desempenho dos alunos é avaliado de acordo com os critérios preestabelecidos, tais como a organização, a postura durante a prática, o método utilizado e o cumprimento da prática.

A etapa de prática individual está relacionada aos elementos do pensamento computacional de (i) decomposição, ao passo que os alunos se organizam para realizar tarefas menores e de forma gradativa; (ii) reconhecimento de padrões, através do planejamento de parte das atividades que são semelhantes e não precisam ser repetidas no desenvolvimento do jogo e; (iii) algoritmo que é a sequência desenvolvida para a criação do jogo.

F. Aplicação

A etapa de aplicação proporciona integração e troca de conhecimentos dos alunos. Sendo assim, após o desenvolvimento do jogo de forma individual os alunos podem trocar conhecimentos no sentido de ajudar o colega a terminar algo que não conseguiu durante a atividade. Esta etapa possui caráter lúdico e de companheirismo na troca de conhecimento. O professor possui a missão de intermediar os alunos, observando se o conhecimento passado na aula foi absorvido e auxiliar na correção dos jogos. Por fim, uma dinâmica é realizada para que os alunos e o professor joguem os jogos desenvolvidos.

IV. PLANO DE ENSINO

O plano de ensino proposto possui carga horária de 40 horas, sendo 20 aulas de duas horas cada. Um ciclo do método de ensino proposto necessita de duas aulas para ser concluído. Na primeira aula de cada ciclo são realizadas as etapas de Aula Lúdica, Idealização, Planejamento e Prática Auxiliada. Já a segunda aula de cada ciclo, inicia com uma pequena revisão do conteúdo abordado seguida pelas etapas de Idealização, Planejamento, Prática Individual e Aplicação.

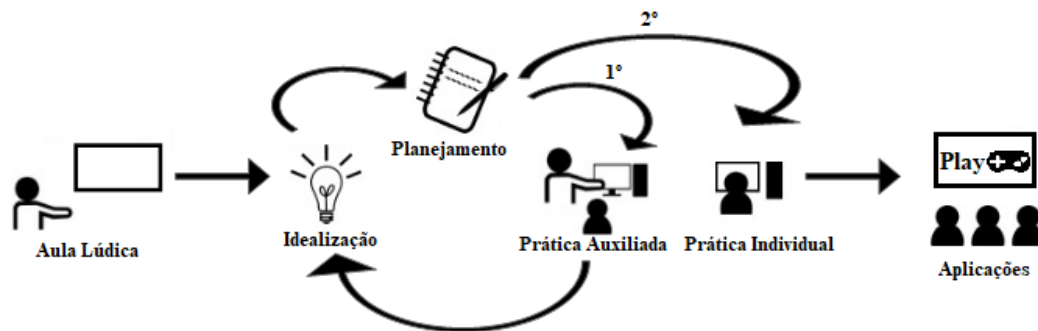


Fig. 1. Fluxograma do método de ensino AI(3P)A. Fonte: autor.

Os tópicos de programação de computadores abordados neste plano de ensino foram (i) as estruturas básicas (sequência, condicionais e laços de repetição), (ii) a utilização da lógica de programação, (iii) programação condicional e (iv) estruturas de repetição.

Na Tabela 1 é apresentado o conteúdo programático do curso. Cada linha da tabela representa um ciclo do método de ensino. Cada ciclo possui o nome da aula, que é nomeado de acordo com um tema, o conteúdo que será apresentado aos alunos, o que será desenvolvido. A Tabela 2 aborda o mesmo planejamento porém indicando o jogo mais adequado para cada aula de acordo com o conteúdo lecionado.

TABELA I
PLANEJAMENTO RESUMIDO DO CURSO. FONTE: AUTOR.

Aula	Conteúdo	Desenvolvimento
1 - Olá Mundo	Conceitos iniciais de programação, jogos eletrônicos e ferramentas Scratch	Compreender conceitos iniciais de desenvolvimento de jogos, de programação e da ferramenta Scratch
2 - Interagindo	Entrada e saída de dados	Aprender a inserir e retornar informações
3 - Será que pode?	Condicional "Se" e operações lógicas	Apresentar as estruturas condicionais, iniciando pela estrutura "Se"
4 - E se não puder?	Condicional "Se - Senão"	Apresentar a estrutura condicional "Se-Senão"
5 - Vamos repetir!	Laço de repetição	Apresentar as estruturas de repetição
6 - Hora de revisar!	Ferramenta Construct e revisão do conteúdo	Apresentação de uma ferramenta mais potente e revisar os conteúdos já aprendidos
7 - Guardando informações!	Variáveis	Aprender a armazenar o conteúdo
8 - Reaproveitando.	Função	Compreender a estratégia de reaproveitar comportamentos
9 - Agora pode?	Eventos	Entender o conceito de eventos
10 - Desafio	Projeto desafio	Desenvolvimento de um projeto com todos os conceitos, com a ideia individual dos alunos, utilizando um tema como base

TABELA II
JOGOS DESENVOLVIDOS EM CADA ETAPA. FONTE: AUTOR.

Aula	Jogo
1 - Olá Mundo	Hello World
2 - Interagindo	Calculando
3 - Será que pode?	T-Rex Running
4 - E se não puder?	Jogo do Zumbi
5 - Vamos repetir!	Pong
6 - Hora de revisar!	Jogo de Naves
7 - Guardando informações!	Flappy Bird
8 - Reaproveitando.	Jogo de Corrida
9 - Agora pode?	Mario Bros
10 - Desafio	Jogo Evolucionário

V. RELATO DE EXPERIÊNCIA UTILIZANDO O MÉTODO DE ENSINO AI(3P)A

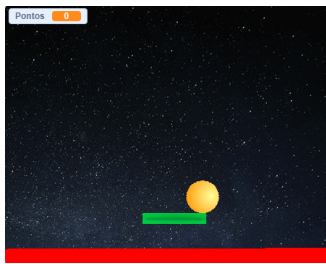
As experiências de aplicação do método de ensino AI(3P)A foram realizadas em dois cenários diferentes, utilizando uma abordagem qualitativa de essência exploratória. Ambas as experiências foram realizadas em alunos do curso de Ensino de Programação Utilizando Jogos Eletrônicos. A primeira turma foi formada por quatro alunos do sexo masculino com faixa etária de oito a doze anos. A segunda turma foi formada também por quatro alunos do sexo masculino, mas com idade de treze a dezesseis anos. Apenas um aluno era da rede de ensino particular, os demais alunos estavam cursando o ensino básico na rede pública de educação.

As atividades foram desenvolvidas no segundo semestre de 2019, totalizando uma carga horária de 40 horas, divididas em 20 aulas de duas horas cada, ocorridas no turno da manhã, uma vez por semana aos sábados. A primeira turma iniciava a aula às 8:20 e encerrava às 10:20 e a segunda turma iniciava às 10:30 e encerrava às 12:30. As aulas aconteciam em laboratórios equipados com computador, lousa e projetor. Os alunos compareceram em todas as aulas e realizaram todas as atividades do plano de ensino proposto.

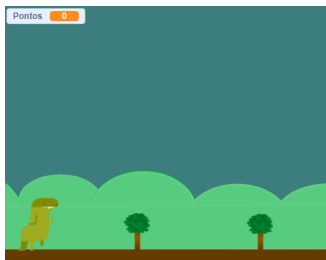
A. Atividades Desenvolvidas

As atividades abordadas no curso seguiram o planejamento supracitado. Em cada aula o método de ensino AI(3P)A foi seguido. Dessa forma, um jogo foi desenvolvido pelos alunos em cada aula durante a prática auxiliada, alguns dos jogos são apresentados na Fig. 2. Os jogos propostos utilizam os

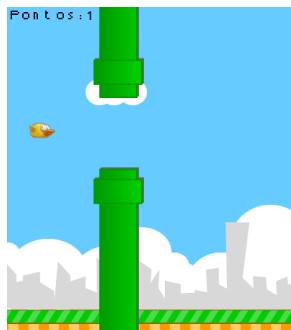
conceitos de programação lecionados na etapa de aula lúdica. Por exemplo, o jogo *Pong* (Fig. 2 (a)) utiliza o conceito de laço de repetição, pois acumula pontos para o jogador sempre que o objeto bola encostar na plataforma. Além disso, o desenvolvimento do jogo utiliza o conceito de condicional aprendido na aula anterior, quando verifica se a bola encostou na plataforma.



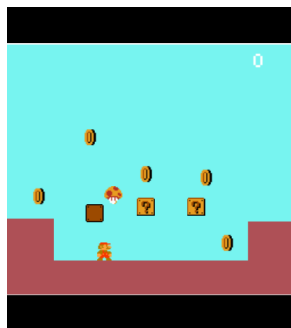
(a) Jogo Pong



(b) Jogo T-Rex Running



(c) Jogo Flappy Bird

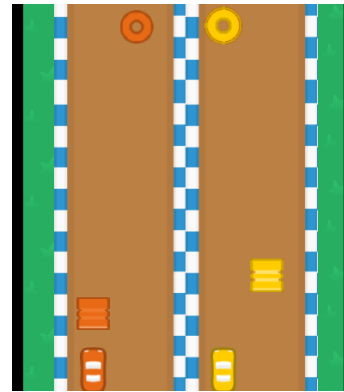


(d) Jogo Mario Bros

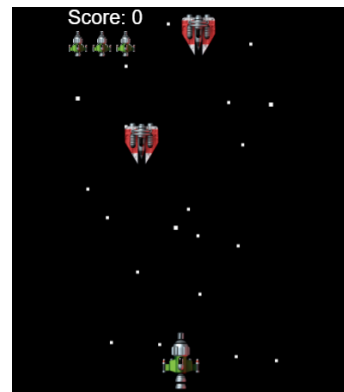
Fig. 2. Jogos propostos pelo professor e desenvolvidos em conjunto com os alunos. Fonte: autor.



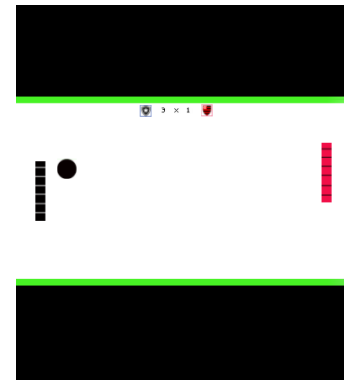
(a) Jogo de Zumbis.



(b) Jogo Go Cars



(c) Jogo Batalha de Naves



(d) Jogo de Futebol

Fig. 3. Jogos propostos e desenvolvidos pelos alunos. Fonte: autor.

Na etapa de prática individual os alunos devem idealizar e planejar um novo jogo que utilize os conceitos aprendidos. Alguns dos jogos desenvolvidos são apresentados na Fig. 3. Esta etapa estimula a criatividade dos alunos de aplicar o novo conceito com um jogo eletrônico.

Entre os jogos desenvolvidos nessa etapa, podemos destacar o jogo *Go Cars* (Fig. 3 (b)). Este jogo foi desenvolvido na última aula que possui o desafio de utilizar todos os conceitos aprendidos dentro do tema evolução. Este tema trata do desenvolvimento de um jogo de evolução cognitiva onde o jogador deve controlar diferentes carrinhos de forma gradativa, iniciando com um carro e aumentando até quatro carros. Para vencer o jogador precisa desviar de todos os obstáculos e coletar itens, se algum desses objetivos não for cumprido o jogador perde.

A última etapa do método AI(3P)A é a de Aplicações, onde os alunos reúnem para auxiliar quem ainda não terminou seu jogo e posteriormente jogar os jogos desenvolvidos. Nessa etapa podemos notar o entusiasmo dos alunos em apresentar o seu jogo para os colegas, mesmo aqueles que possuíam dificuldades de terminar o jogo, não tiveram problemas em pedir ajuda para finalizar.

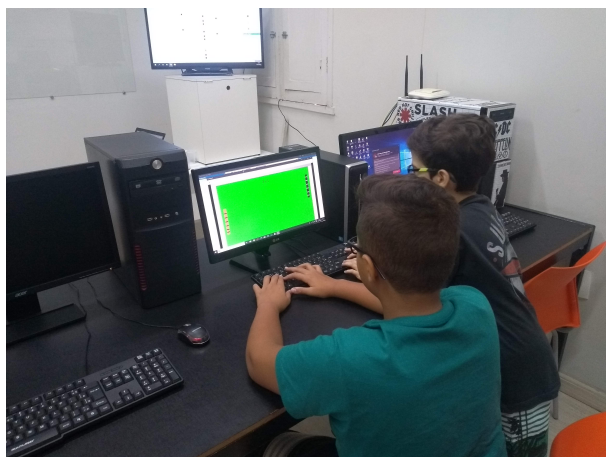


Fig. 4. Alunos jogando um dos seus jogos na etapa de aplicação. Fonte: autor.

B. Discussão e Resultados

O desenvolvimento do conhecimento de lógica de programação para crianças e adolescentes, pode beneficiar no desenvolvimento de soluções para problemas, estimular o trabalho em equipe e o raciocínio lógico. Essas capacidades são muito importantes na vida escolar das crianças embora os métodos tradicionais de ensino de lógica de programação podem desmotivar o interesse em estudar programação.

Este trabalho possibilitou avaliar como o método de ensino AI(3P)A auxilia no ensino de lógica de programação, utilizando ferramentas de desenvolvimento de jogos eletrônicos como ferramenta pedagógica, no cenário de uma escola de capacitação, com duas turmas compostas de faixas etárias diferentes.

A avaliação da percepção do aluno (Fig. 5) foi coletada utilizando um formulário que aborda a visão pessoal sobre as atividades desenvolvidas sob três condições: satisfação, motivação e controle. A escala utilizada no formulário foi de totalmente satisfeito, motivado e controlado (100%) à totalmente insatisfeito, desmotivado e pouco controlado (0%). Os formulários foram aplicados no final de cada aula, assim totalizando 40 repostas por turma. Na primeira turma, com faixa etária de oito a doze anos, a média das respostas dos alunos foram de 95% para satisfação, 80% para motivação e 75% para controle. Já a segunda turma, com faixa etária de treze a dezesseis anos, as respostas foram de 90% para satisfação, 95% para motivação e 80% para controle.



Fig. 5. Formulário de percepção preenchido pelos alunos. Fonte: autor.

Assim, o método de ensino proposto possibilitou que a aprendizagem de conceitos de lógica e programação fossem atrativos e divertidos devido a apresentação utilizando uma forma lúdica e com ferramentas que atraem a atenção de crianças e adolescentes. Esse fato é correlativo com os resultados do formulário autoavaliativo sobre o sentimento dos alunos. Os dados coletados, em média, das duas turmas possui níveis de satisfação, motivação e controle positivos respectivamente em 92,5%, 87,5% e 80%.

Outro questionário foi aplicado no final de cada aula com o objetivo de avaliar as atividades desenvolvidas. O formulário consistiu de três avaliações discursivas (BOM, RUIM e MELHORAR). Seu objetivo é coletar informações que poderiam ser utilizadas para melhorar as atividades propostas ou auxiliar em alguma dúvida não solucionada na sala de aula. Neste questionário, as atividades foram avaliadas pelos alunos como BOAS e destacaram a utilização de jogos eletrônicos e as práticas como razão que favoreceu a satisfação pelo o curso.

VI. CONSIDERAÇÕES FINAIS

Este artigo apresentou o método de ensino AI(3P)A, bem como as experiências de aplicação do método no ensino de programação de computadores utilizando ferramentas de desenvolvimento de jogos eletrônicos. O objetivo do método é proporcionar o ensino de conceitos de programação de maneira divertida, sendo uma prática cativante para os alunos. Os resultados obtidos nesse trabalho foram retirados da avaliação das atividades desenvolvidas pelos alunos e dos formulários de autoavaliação. Os resultados foram positivos e mostraram que o método utilizado no ensino de lógica de programação pode ser um fator primordial no aprendizado e na motivação

dos alunos. O método proposto seguiu os elementos do pensamento computacional, assim contribuindo para o desenvolvimento do mesmo e para o aprendizado de programação de computadores, pois possui uma abordagem lúdica, prática e evolutiva dos conceitos. Como atividades futuras a serem desenvolvidas, pretendemos avaliar como o método de ensino AI(3P)A impacta na vida acadêmica dos alunos, por exemplo, nas matérias da exata que utilizam lógica. Busca-se verificar também se este curso ao longo prazo pode incentivar os alunos a ingressar em um curso superior da área tecnológica.

REFERENCES

- [1] T. Bell, IH Witten, and M. Fellows, Computer science unplugged—ensinando ciência da computação sem o uso do computador, 2011.
- [2] R. I. Casarotto, G. Bernardi, A. Z. Cordenonsi, and R. D. Medina. Logirunner: um jogo de tabuleiro como ferramenta para o auxílio do ensino e aprendizagem de algoritmos e lógica de programação. *RENOTE-Revista Novas Tecnologias na Educação*, 2018, pp. 16(1).
- [3] I. Dantas, J. Neto, L. Silva, L. Neto, D. Lima, P. Scaico, and T. Costa. Ensino de lógica de programação no ensino fundamental utilizando o jogo robotizen: um relato de experiência. In *Anais do XXVII Workshop sobre Educação em Computação*, 2019, pp. 51–60.
- [4] A. C. Ferreira, J. Santos, R. Silva, A. T. R. Oliveira, D. Zabet, D. Abdalla, and E. Matos. Helloworld: relato de experiência de um curso de iniciação à programação. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, 2016, pp. 1306.
- [5] F. P. Mota, N. F. A. Ribeiro, L. Emmendorfer, P. Butzen, K. S. Machado, and D. F. Adamatti. Desenvolvendo o raciocínio lógico no ensino médio: uma proposta utilizando as ferramentas scratch. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, 2014, pp. 377.
- [6] M. Prensky. Digital natives, digital immigrants. *On the horizon*, 2001.
- [7] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, et al. Scratch: programming for all. *Communications of the ACM*, 2009, pp. 52(11):60–67.
- [8] R. Santos, B. Sousa, A. Raiol, P. Cerqueira, and F. Bezerra. Uma proposta de método de ensino e relatos de experiências com a robótica educacional. In *Anais do XXVII Workshop sobre Educação em Computação*, 2019, pp. 111–120.
- [9] S. Souza and T. Castro. Investigação em programação com scratch para crianças: uma revisão sistemática da literatura. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, 2016, pp. 1078.
- [10] G. A. Viana and C. S. Portela. O uso de softwares educativos para introdução de lógica de programação no ensino de base e superior. *Informática na educação: teoria prática*, 2019, pp. 22(1).
- [11] J. M. Wing. Computational thinking. *Communications of the ACM*, 2006, pp. 49(3):33–35.
- [12] J. M. Wing. Research notebook: Computational thinking—what and why? the link. *The magazine of the Carnegie Mellon University School of Computer Science*, 2011.