

Aulas de programação durante a Pandemia: vencendo o Desafio da Distância

Allan M. Cidreira¹, Amanda P. Manso¹, Iara M. Henriques F. e Silva¹, Lorenzo M. Correia¹, Mayk T. Bezerra¹, Renato Bueno²

Departamento de Computação
Universidade Federal de São Carlos (UFSCar) - São Carlos, SP - Brasil

{allan, amandamanso, iaramhfs, lorenzomaia, mayk}@estudante.ufscar.br,
renato@ufscar.br

Abstract. *This paper describes the challenges and found solutions in the proffer of programming classes to 13 to 15 years old juveniles, based on an applied project by scholarship holders in PET BCC. The main goal was to teach and present the algorithm thinking to kids, in a contemporary and attractive way. The project presented the teaching of Python and used online platforms like Scratch and Replit, using different ways of non-presential evaluation. The classes happened during the year of 2021, having, as main scenery, the coronavirus pandemic.*

Resumo. *Este artigo relata os desafios e soluções encontradas no oferecimento de aulas de programação para jovens de 13 a 15 anos, baseado em um projeto aplicado por bolsistas do PET BCC. O objetivo principal foi ensinar e apresentar o pensamento algorítmico para os jovens, de forma contemporânea e atrativa. O projeto apresentou o ensino de Python e utilizou ambientes online como o Scratch e o Replit, utilizando diferentes ferramentas de avaliação não-presencial. As aulas aconteceram durante o ano de 2021, tendo, como cenário principal, a pandemia do novo coronavírus.*

1. Introdução

Nos últimos anos, o crescente envolvimento dos jovens com novas tecnologias gerou a denominação da geração atual de geração *online*, segundo Spirizzi et al. (2017). Com o maior contato com a internet, termos como computação e algoritmo podem aparecer em diversos cenários, mas seus significados não estão tão presentes. Por definição, um algoritmo é uma sequência de passos lógicos que levam a realização de uma tarefa, como é citado em *Algorithm* (2021).

Além do crescente uso de tecnologia, o período de 2020 e 2021 foi marcado pela pandemia do novo coronavírus (COVID-19), onde as escolas haviam sido fechadas e o distanciamento social foi implantado. Desta forma, conforme Fogaça et al. (2020), indivíduos em distanciamento social são mais suscetíveis ao estresse causado pela condição e, como consequência da privação social, os indivíduos têm apresentado aumentos de transtornos de ansiedade, transtornos depressivos e alterações na qualidade do sono.

Tendo estes pontos como motivadores, foi idealizado um projeto pelo Programa de Educação Tutorial (PET) do curso de Bacharelado em Ciência da Computação da UFSCar, as Aulas Virtuais de Programação, com a finalidade de apresentar conceitos sobre o pensamento algorítmico para jovens de forma didática e atrativa, além de ajudá-los a aprender de modo leve, tranquilo e divertido, fornecendo apoio durante o período de distanciamento social.

¹ Bolsista do Programa de Educação Tutorial - PET-BCC - Universidade Federal de São Carlos

² Tutor do Programa de Educação Tutorial - PET-BCC - Universidade Federal de São Carlos

2. Divulgação e Dados

Um mês antes de iniciarem as aulas, durante as primeiras semanas de Março/2021, a equipe do curso de Aulas Virtuais de Programação iniciou as inscrições através de um formulário do Google Forms de manifestação de interesse.

A divulgação foi realizada por meio de redes sociais do PET BCC, como a página oficial do Facebook, por listas de e-mail dos professores da universidade, requisitando a divulgação caso conhecesse pessoas que se encaixassem no perfil, e através de pesquisa nas escolas públicas próximas da região de São Carlos. A divulgação pelas redes sociais e e-mails, além do formulário, continham um banner de divulgação, conforme exibido na Figura 1.

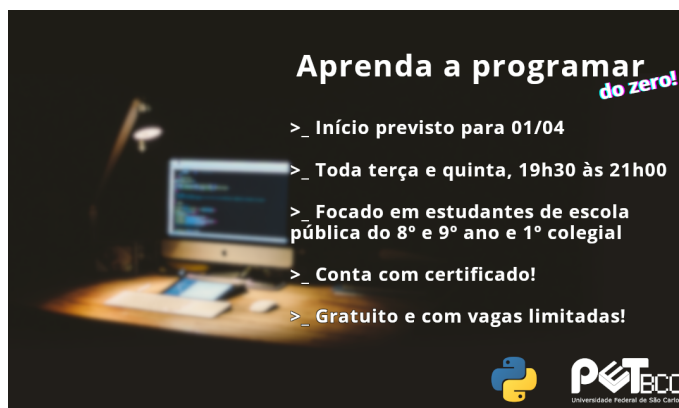


Figura 1: Banner utilizado para a divulgação por redes sociais

No total, foram recebidas 156 manifestações de interesse pelo formulário, sendo 94.2% estudantes de escola pública, com idades entre 12 a 17 anos, distribuídos desde o 7º ano do fundamental II até o 3º ano do ensino médio. Além disso, foram recebidas algumas manifestações de professores e adultos. O número de inscrições foi visto como sucesso pela equipe, atribuído à indicação e recomendação de professores de escolas, que o grupo entrou em contato para a divulgação. Informações sobre como os inscritos souberam do curso são apresentados na Figura 2.

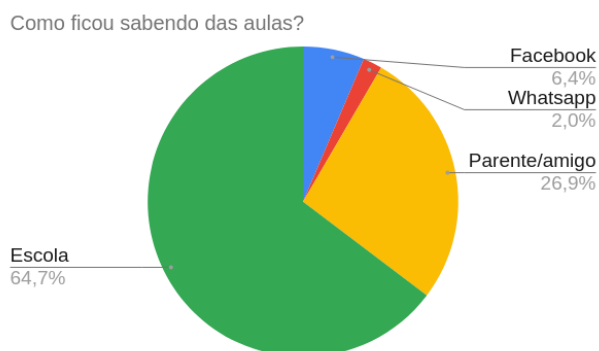
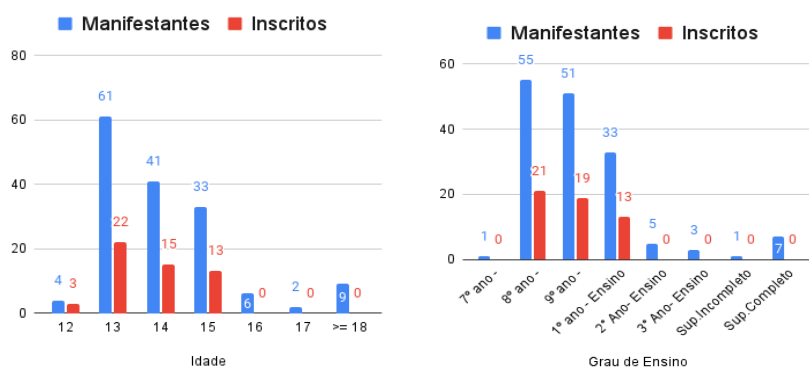


Figura 2: Gráfico de proporção de como os manifestantes souberam do curso

Devido ao alto número de interessados, a equipe decidiu, então, por realizar as seguintes mudanças: criar um segundo formulário de interesse, buscando afunilar a quantidade de interessados; cumprir com o esperado inicialmente de público alvo, aprovando apenas alunos de 8º ano à 1º ano do Ensino Médio de escolas pública e, por fim, requisitar mais pessoas do grupo PET BCC para atuarem como professores, de forma a assistir mais alunos durante o curso.

Desta forma, após realizado o levantamento de interesse, outro formulário foi enviado aos interessados para confirmação da inscrição no curso. Com o retorno e devido à disponibilidade de professores, foram selecionados 53 alunos, cerca de 34% do total de pessoas que se manifestaram no primeiro formulário, que se encaixavam na faixa etária e que eram de escolas públicas, conforme exibidos nas Figuras 3 e 4. Os alunos foram distribuídos em 4 turmas com diferentes horários. Pela distribuição de alunos, a equipe decidiu que cada turma teria 2 professores fixos para melhor dinâmica durante as aulas: um responsável pela aula expositiva e o outro atuando como monitor, apoiando a aula e tirando dúvidas no chat da turma.



Figuras 3 e 4: Relação da idade e grau de ensino dos manifestantes e inscritos no curso

O curso foi realizado em 14 semanas, iniciando em Abril, com uma apresentação de abertura com a presença de todos os alunos inscritos, e terminando na metade de Julho de 2021, com um encerramento e apresentação do projeto final do curso.

3. Ferramentas

Para a inicialização dos alunos no pensamento algorítmico foi utilizada a ferramenta Scratch³, desenvolvida pelo *Massachusetts Institute of Technology* (MIT). Como apresenta Marji (2014), a plataforma permite criar projetos interativos e visuais, incluindo histórias animadas, jogos, simulações e muitos outros, promovendo habilidades de resolução de problemas.

Após a etapa inicial, foi utilizada a linguagem de programação Python para continuar o ensino de algoritmos. Ela foi escolhida devido à sua popularidade e por ter a característica de ser uma linguagem de alto nível. Além disso, Briggs (2013) mostra que a linguagem provê ferramentas úteis para programadores iniciantes, como um código fácil de ler e um *shell* interativo onde é possível compilar programas e vê-los funcionar. A linguagem Python foi apresentada através do ambiente online Replit⁴.

Para as aulas e para comunicação entre os alunos e professores, foi escolhida a plataforma online Google Meet, de fácil acesso e gratuita. Para a realização do trabalho final, foi escolhida a plataforma Discord, onde, devido à possibilidade de criar vários canais de voz e os professores poderem alternar entre eles, as aulas de desenvolvimento do trabalho puderam ser realizadas em um ambiente mais interativo.

4. Planejamento: diretrizes e importância da motivação

Para apresentar o pensamento algorítmico às crianças, a equipe considerou, como pontos importantes para a criação do material e organização das aulas, os seguintes pontos: como trazer a computação mais próxima à realidade dos alunos? Como adaptar o ensino para o momento atual, de pandemia do novo coronavírus?

³ <https://scratch.mit.edu/about/>

⁴ <https://replit.com/>

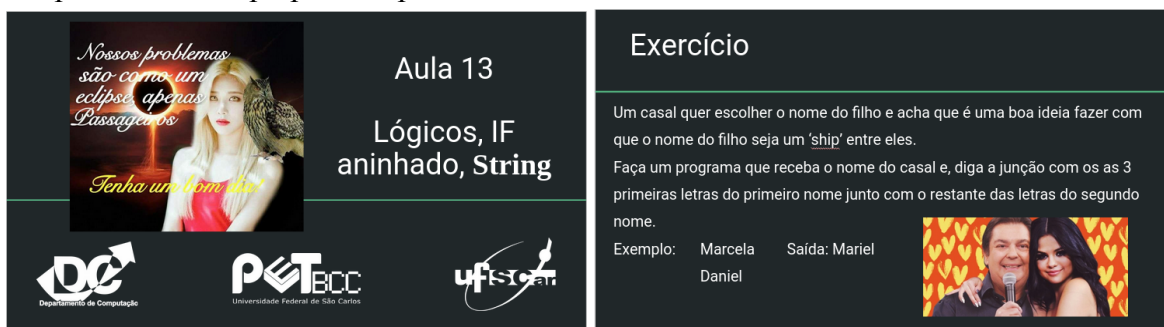
Para abranger o primeiro ponto, o grupo considerou como é o dia-a-dia das crianças: são jovens com acesso à internet, que passam o tempo de lazer em redes sociais e conversando com amigos pela internet [6]. Com o objetivo de despertar o interesse em participação desse público, de acordo com de Jesus Oliveira et al. (2019), a equipe decidiu utilizar ‘Memes’ de redes sociais como elementos didáticos, estando presentes tanto no material produzido quanto na didática dos professores. Assim, a equipe planejou: material das aulas com foco, majoritariamente, visual, e apresentar o pensamento algorítmico com elementos cotidianos e bem-humorados.

Quanto ao segundo ponto, refletindo sobre a instabilidade e criticidade do período pandêmico, foi necessário adaptar o modelo para que as aulas fossem, além de um momento de aprendizado, um momento também de lazer. Ademais, o grupo pensou em possíveis imprevistos durante o período, como instabilidades de conexão de internet ou problemas com equipamentos eletrônicos (como computador, celular, etc.). Desta forma, considerou-se os seguintes pontos: aulas rápidas, de, no máximo, 1 hora e 30 minutos; não-obrigatoriedade da participação em todas as aulas, mantendo-se 75% de presença para obtenção do certificado de conclusão do curso e disponibilização da gravação das aulas, além da não-obrigatoriedade de interação por microfone ou webcam (apesar de serem incentivadas para a interação com os professores e demais colegas).

5. Preparação do Material

Após alinhar os principais objetivos a serem atendidos com as aulas de programação, a equipe preparou o plano de ensino das aulas. Como base, resolveu-se dividir o curso em três etapas: lógica de Programação - utilizando a plataforma Scratch; introdução à Python - utilizando o Replit e consolidação de conhecimento - através do projeto final.

O grupo, então, compreendeu que o material preparado em formato de apresentação (*slides*) seria um dos principais meios de tornar as aulas confortáveis e motivadoras para o público. Desta forma, a preparação do material consistiu nas seguintes etapas: adequação do *layout* padrão de identidade das Aulas de Programação do PET BCC; criação de exercícios adequados para serem realizados pelos alunos em aula e fora delas e adequação dos *slides* com o conteúdo, colocando imagens e *gifs* de assuntos cômicos e cotidianos de redes sociais e relacionando-os com o conteúdo a ser apresentado em aula. Nas Figuras 8 e 9, alguns exemplos dos slides preparados para as aulas.



Figuras 8 e 9: Slides com adequação de imagens para o público.

6. Realização das Aulas

Além da apresentação do conteúdo, os professores asseguravam um espaço seguro para a comunicação, motivando os alunos a elucidar dúvidas e demonstrando a importância de responder todas as perguntas, mesmo que de conteúdo anterior. De forma a reafirmar a segurança do espaço, no decorrer das aulas, foi implementado um formulário anônimo para os alunos, para o levantamento de perguntas em cada uma das aulas e em cada turma.

Em alguns momentos do curso, foram realizadas atividades além da aula expositiva, visando evitar a evasão de alunos, mantê-los interessados no curso e avaliar o aprendizado. As atividades foram: Dojo entre alunos e o uso da plataforma online Kahoot para a realização de perguntas sobre o conteúdo. Ambos métodos são apresentados na seção a seguir.

Para finalizar o curso, foram realizadas aulas especiais e uma apresentação final dos alunos, apresentando os jogos desenvolvidos como projeto final.

6.1. Dojo

O Dojo é uma atividade onde duas pessoas se juntam para tentar resolver um determinado problema de computação, onde uma escreve o código (piloto) e a outra passa a lógica para resolver o problema (co-piloto). O principal objetivo é criar um ambiente colaborativo e não competitivo, propiciando que os participantes aprendam a partir da troca de experiências em programação, como apresenta Sato et al. (2014). Para avaliar se a atividade estava sendo positiva na visão dos alunos, foi aplicado um formulário de *feedback*, coletando opiniões dos alunos sobre a atividade.

Após a aplicação do método, foi possível entender as dificuldades dos alunos em resolver os problemas e se o conteúdo das aulas foi compreendido. Analisando o formulário de *feedback*, as opiniões sobre o dojo foram discrepantes. Entre os principais problemas apresentados pelos alunos, destacava-se a timidez e desconforto em interagir com os professores e colegas pelo microfone ou apresentando a tela. Em algumas turmas, quando questionados sobre o desejo de ter o método novamente em aulas futuras, as opiniões dos alunos apresentavam-se divergentes entre si, onde metade da turma não gostaria de ter novamente (notas abaixo de 5) e a outra metade gostaria (notas acima de 6), conforme exibido na Figura 10. Assim, os professores do curso optaram por não utilizar mais este método nessas turmas e mudar de estratégia, buscando outras formas de acompanhar os alunos.

De 0 a 10, quanto você gostaria de ter um dojo em próximas aulas?

9 respostas

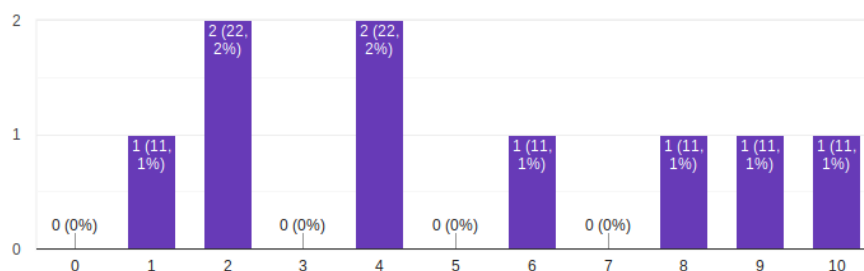


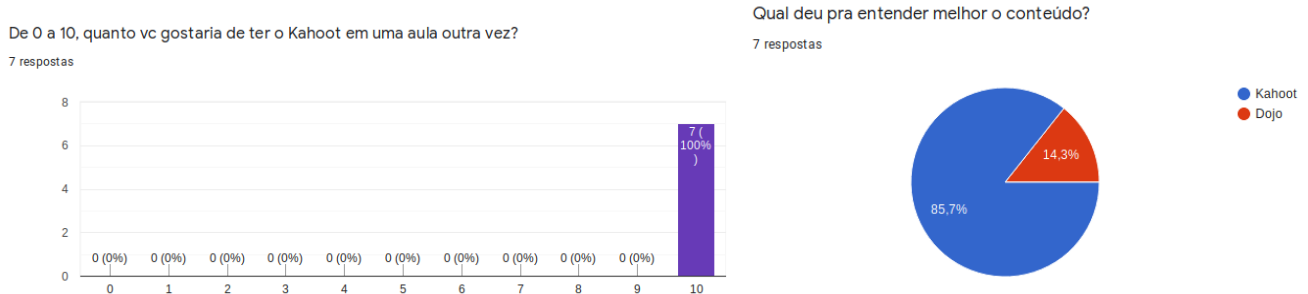
Figura 10: Gráfico extraído do formulário de *feedback* do Dojo.

6.2. Kahoot

O Kahoot foi outra atividade realizada durante o curso, com o objetivo de avaliar e acompanhar como estava o aprendizado dos alunos de forma anônima. A plataforma *online* é uma ferramenta onde os professores podem montar diversos *quizzes*, com diferentes opções de respostas (como verdadeiro ou falso, alternativas, etc.). Os *quizzes* tinham um tempo para cada pergunta e a resposta era anônima, somente com um pseudônimo escolhido por cada aluno. Segundo Junior (2017), através das respostas, os professores poderiam entender onde os alunos apresentavam maiores dificuldades e retomar pontos importantes do conteúdo.

Foi realizado um formulário de *feedback* do uso do Kahoot, que, de forma geral, obteve grande aceitação por parte dos alunos, mantendo sua utilização até o final do curso.

Quando questionados se gostariam de ter o método aplicado novamente em aulas futuras, a adesão foi de 100% dos alunos, conforme exibido na Figura 11. Além disso, quando questionado qual dos métodos, Kahoot ou Dojo, os alunos acreditam que foi possível elucidar o conteúdo de forma mais concreta, a maior parcela dos alunos escolheu o Kahoot, conforme exibido na Figura 12.



Figuras 11 e 12: Gráficos extraídos do formulário de *feedback* sobre o Kahoot.

6.3. Aulas especiais

Duas aulas especiais foram preparadas para as turmas com o objetivo de incentivar os alunos a continuarem acompanhando o curso. A primeira, que ocorreu na metade do curso, contou com a apresentação de aplicações avançadas do conteúdo de Python. Algumas dessas aplicações foram: Pentest de Redes WIFI, Reconhecimento Facial, construção de sites/blogs, sintetização e reconhecimento de voz, entre outros. As aplicações mostraram aos alunos que diversos elementos presentes no dia-a-dia ou tecnologias que parecem longe de acontecer estão ao alcance do conhecimento que eles teriam contato durante o curso, especificamente sobre linguagem de programação Python.

A segunda aula especial foi preparada para o final do curso, e a sua principal motivação era parabenizar a conclusão do curso e incentivar o ingresso dos alunos ao ensino superior. Chamada de “Como é a vida na universidade?”, foi apresentado como era o espaço físico da Universidade Federal de São Carlos (UFSCar) e do Departamento de Computação, além da apresentação de atividades que eram realizadas no campus e dentro do departamento, projetos de extensão, eventos oficiais, entre outros. Por último, foram apresentadas as formas de ingresso nas universidades e um espaço para responder dúvidas.

6.4. Jogos

Conforme apresentado por Baibak e Agrawal (2007), os jogos são comumente utilizados no aprendizado de algoritmos e linguagens de programação, e tiveram um papel importante neste curso como forma de fixação do conteúdo aprendido.

Como forma de incentivá-los a praticar a programação e, também, como uma forma de avaliação final, foram apresentados alguns jogos simples que poderiam ser feitos com o aprendizado adquirido ao longo dos meses, como Jogo da velha, Jogo da Forca, Jogo da memória, RPG Textual (modelo de jogo textual com tomadas de decisões para diferentes ramificações de histórias), entre outros.

Ao demonstrar como fazer os jogos, os professores incentivaram os alunos a mistura-los com um assunto de seu interesse. Uma estratégia da equipe para que os alunos, além de praticarem a programação, praticarem, também, a imaginação e a criatividade, possibilitando aos estudantes enxergarem, no dia-a-dia de suas atividades, os conceitos aprendidos durante o curso. Nas apresentações dos jogos, como, por exemplo, o RPG Textual, onde foram criadas histórias de aventura, terror, entre outras, ficou evidente aos professores e

alunos presentes que a estratégia funcionou e foi implementada na criação dos jogos. Na opinião da equipe, o resultado dos jogos produzidos ficou acima do esperado.

7. Conclusão

O projeto de aulas de programação atendeu cerca de 50 alunos. Desde o início das aulas, os alunos demonstraram interesse em conhecer mais sobre a área da computação e alguns relataram que gostariam de seguir profissionalmente na área. Os discentes eram participativos nas aulas e fora delas, realizando as atividades propostas e buscando aprender além do que era apresentado, contatando os professores, ainda, fora das aulas, por mensagens de texto ou chamadas externas, onde os alunos de conversavam sobre o conteúdo apresentado ou interagirem entre si, por jogos online ou conversas cotidianas. Os professores perceberam que, mesmo à distância, os jovens pareciam gostar das aulas, da turma e dos professores, considerando-os como colegas de classe.

O grupo percebeu uma boa retenção de alunos durante todo o período do curso, com poucas desistências, visto que, ao final do curso, 34 dos 53 alunos que concretizaram a inscrição concluíram o curso, correspondendo a 64,15% dos inscritos. A equipe atribui as desistências a questões específicas relacionadas à Pandemia do COVID-19, como a divisão de atenção e tempo das crianças tanto quanto ao curso de aulas de programação quanto às atividades escolares concomitantes.

É válido ressaltar, também, a participação dos alunos na Olimpíada Brasileira de Informática (OBI). Aqueles que demonstraram interesse em participar obtiveram bons resultados, demonstrado pela classificação para fases posteriores. Os professores dedicavam-se ao atendimento especializado para a Olimpíada em horários externos às aulas, incentivando a participação dos jovens em eventos e a continuarem aprendendo sobre programação.

8. Referências

- Spizzirri, R. C. P., Wagner, A., Mosmann, C. P., & Armani, A. B. (2017). Adolescência conectada: Mapeando o uso da internet em jovens internautas. *Psicologia Argumento*, 30(69).
- Britannica, The Editors of Encyclopaedia. "algorithm". Encyclopedia Britannica, 6 May. 2021, <https://www.britannica.com/science/algorithm>. Accessed 19 September 2021.
- Fogaça, P. C., Arossi, G. A., & Hirdes, A. (2021). Impacto do isolamento social ocasionado pela pandemia COVID-19 sobre a saúde mental da população em geral: Uma revisão integrativa. *Research, Society and Development*, 10(4), e52010414411-e52010414411.
- Marji, M. (2014). Learn to program with Scratch: A visual introduction to programming with games, art, science, and math. No Starch Press.
- Briggs, J. R. (2013). *Python for kids: A playful introduction to programming*. no starch press.
- Passarelli, B. (2020). Jovens brasileiros em conectividade contínua: estudos e tendências. *Juventude e Políticas Públicas [Internet]*. 27º de janeiro de, 2, 1-16. Porto, C., & Alves, A. L. (2019). Memes de redes sociais digitais enquanto objetos de aprendizagem na Cibercultura: da viralização à educação. *Acta Scientiarum. Education*, 41, e42469-e42469.
- Sato, D. T., Corbucci, H., and Bravo, M. V. (2014). Coding dojo: An environment for learning and sharing agile practices. In Agile Conference, pages 459–464. IEEE.
- Junior, J. B. B. (2017). O aplicativo Kahoot na educação: verificando os conhecimentos dos alunos em tempo real. In *Livro de atas X Conferência Internacional de TIC na Educação—Clallenges* (pp. 1587-1602).

Baibak, T., & Agrawal, R. (2007, June). Programming games to learn algorithms. In *2007 Annual Conference & Exposition* (pp. 12-1191).