

Strategies Focused on the Teaching of Programming Logic: A Systematic Review of Brazilian Literature

Deborah Godoy Martins Corrêa
Instituto de Ciência e Tecnologia
Universidade Federal de São Paulo
São José dos Campos, Brasil
deborah.correa@unifesp.br

Tiago de Oliveira
Instituto de Ciência e Tecnologia
Universidade Federal de São Paulo
São José dos Campos, Brasil
tiago.oliveira@unifesp.br

Denise Stringhini
Instituto de Ciência e Tecnologia
Universidade Federal de São Paulo
São José dos Campos, Brasil
dstringhini@unifesp.br

Abstract—This article aims to present the results obtained from the systematic of the Brazilian literature on teaching and learning the discipline of Programming Logic (LP), considering the articles published in the Brazilian Journal of Informatics in Education (RBIE), in the annals of the Brazilian Symposium on Informatics (SBIE), in the annals of the Computer Science Workshops (WIE), in the Brazilian Congress of Informatics in Education (CBIE) and in the Journey of Updating in Computer Science at School (JAIE) between 2012 and 2017. From this mapping the strategies used in teaching LP in Basic Education, High School and Higher Education were analyzed, and was verified that the subject remains a great challenge to be overcome. Additionally, this work investigates if Multiple Intelligences are being considered in the adoption of teaching techniques - the goal is to establish a future work on this subject. Results show that Scratch and Games are the most cited in Brazilian literature and that Multiple Intelligences are still an open research field in this context.

Keywords—programming logic, multiple intelligences, games, robotics, Scratch

I. INTRODUÇÃO

Pesquisadores e docentes têm desenvolvido diversas estratégias e abordagens para o ensino da disciplina de Lógica de Programação, porém os resultados não têm se mostrado expressivos em relação à melhora do nível de aprendizado dos estudantes. Esta disciplina é básica na área da computação, e a complexidade dos conteúdos geram desmotivação e altos índices de retenção e evasão [1].

As principais causas apontadas na literatura estão centradas nos estudantes que acessam o ensino superior sem a base matemática necessária, apresentam baixa capacidade de abstração e falta de habilidade para entender e resolver problemas, assim como dificuldade em descrever os processos envolvidos [2].

Observando o desempenho dos ingressantes no curso de Bacharelado em Ciência e Tecnologia no período Integral do Instituto de Ciência e Tecnologia (ICT) da Universidade Federal de São Paulo (UNIFESP) que cursaram a disciplina oferecida no 1º semestre de 2011 a 2017, verifica-se que 714 (51,3%) foram aprovados e 678 (48,7%) reprovados (Fig.1).

Segundo Raabe e Silva [3] aprender a programar não é uma tarefa fácil e independentemente da linguagem escolhida exige dos alunos muito foco, determinação e abstração e dos docentes uma forte demanda de interação a fim de atender, acompanhar, mediar e avaliar os alunos, o que é dificultado pelo elevado número de alunos por turma e pelos distintos graus de dificuldade que apresentam.

No Japão o ensino de programação inicia-se na Educação Básica visando uma melhoria do raciocínio lógico e da

capacidade de resolver problemas reais [4]. A disciplina foi incorporada à grade curricular na Estônia (2012), Inglaterra (2014), Austrália (2015) e nos Estados Unidos após Barack Obama (2013) afirmar que "...saber programar um computador hoje é tão básico quanto saber ler, escrever e fazer contas e deve ser ensinado em todas as escolas".

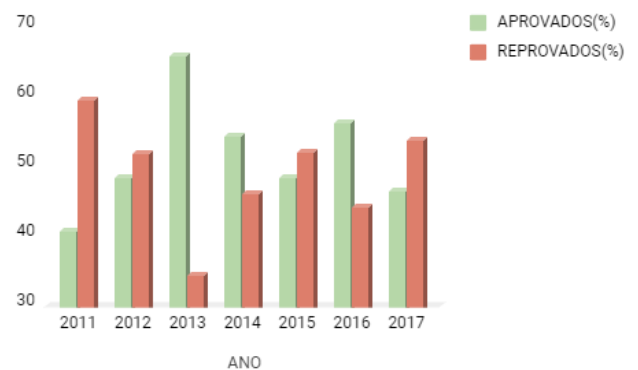


Fig. 1. Aproveitamento dos ingressantes do ICT. Fonte: ProGrad/Unifesp

Uma iniciativa global para a democratização do ensino da programação é a plataforma Code.org utilizada por mais de 18 milhões de estudantes. No Brasil, a partir de 2014, surgiram na iniciativa privada a Mad Code e a Let's Code Academy com o mesmo foco [5,6,7].

Diversas ferramentas de ensino para linguagens de programação são utilizadas para motivar os estudantes como, por exemplo, o desenvolvimento de ambientes gamificados, o software *Scratch* - que utiliza blocos lógicos e itens de som e imagem para desenvolver histórias interativas, jogos e animações, e a robótica educacional, que segundo Maia [8] "é uma prática envolvendo hardware e software, onde a lógica é inerente na montagem e programação dos robôs envolvendo normalmente problemas do mundo real que estimulam o aprendizado de conceitos intuitivos".

Uma questão interessante a considerar em relação à escolha de conteúdos educacionais refere-se a Teoria das Inteligências Múltiplas (TIM) de Howard Gardner [9] que define inicialmente que todas as pessoas são dotadas de sete inteligências: linguística, lógico-matemática, musical, espacial, corporal cinestésica, interpessoal e intrapessoal, e posteriormente acrescentou a inteligência naturalista.

Gardner conceitua a inteligência como sendo "um potencial biopsicológico para processar informações que pode ser ativado num cenário cultural para solucionar problemas ou criar produtos que sejam valorizados numa cultura".

Para Walter [10], a Teoria das Inteligências Múltiplas aplicada ao processo ensino-aprendizagem proporciona aos

docentes conhecer as inteligências mais expressivas de seus alunos, assim como estimular as habilidades ou inteligências que não se apresentam plenamente desenvolvidas.

Na literatura há inúmeros artigos sobre o ensino-aprendizagem de Programação, e uma opção para a coleta e análise é através da revisão sistemática da literatura, que segundo Kitchenham [11] é “uma forma de estudo secundário que utiliza uma metodologia bem definida para identificar, analisar e interpretar todas as evidências disponíveis relacionadas a uma questão de pesquisa específica de uma maneira que seja imparcial e (até certo ponto) repetível”.

O objetivo desta revisão é identificar os resultados das estratégias mais utilizadas nos últimos cinco anos no ensino de Programação no Brasil, no contexto dos ensinos fundamental, médio e superior, e fornecer subsídios para responder a pergunta: A partir do mapeamento de métodos e técnicas utilizados no processo de ensino-aprendizagem da disciplina de Programação, que produto de inovação poderá influenciar positivamente o desempenho dos estudantes a partir da estimulação de suas Inteligências Múltiplas?

II. MÉTODO DE PESQUISA

As revisões sistemáticas têm como objetivos auxiliar a formulação de um problema, e encontrar suas possíveis respostas. Para que isto aconteça é necessário saber onde estão os artigos de interesse, como estão organizados e principalmente, como tirar o máximo proveito deles.

Segundo Kitchenham [11], a revisão sistemática consiste em uma metodologia rigorosa e confiável, que pode ser conferida, auditada ou continuada, e que proporciona resultados mais amplos, consistentes e confiáveis. Auxilia na identificação de lacunas nas pesquisas que foram realizadas sobre o assunto e sugere a necessidade de novas pesquisas.

A revisão sistemática proposta por este trabalho foi executada conforme o protocolo de planejamento, execução e análise dos artigos encontrados na literatura brasileira (Fig.2).

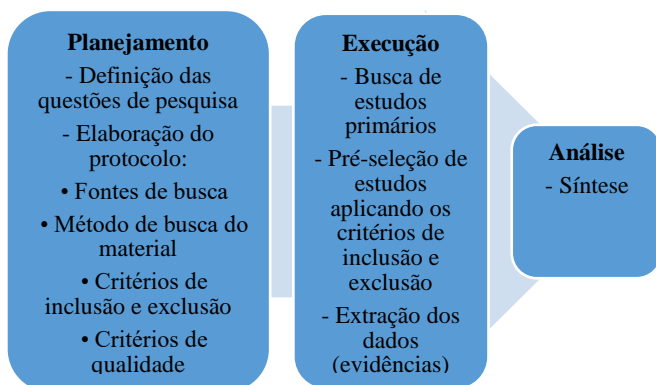


Fig. 2. Protocolo desta revisão sistemática

A. Questões de Pesquisa

A prática baseada em evidências (PBE) propõe que os problemas sejam decompostos e a seguir organizados de forma a facilitar e maximizar o alcance da pesquisa [12].

Para auxiliar na elaboração das questões de pesquisa e criar uma estrutura lógica para a revisão bibliográfica, a PBE

propõe a utilização da estratégia PICO, acrônimo para População, Intervenção, Comparação e Outcomes (resultados), e os componentes relacionados ao foco do tema estão esquematizados na Fig.3.

P Estudantes da Educação Básica e Superior	I Detectar se as pesquisas estão considerando a estimulação das Inteligências Múltiplas
C Analisar as pesquisas mais relevantes que propõem intervenções facilitadoras do aprendizado	O Reconhecer as metodologias e estratégias que favorecem o aprendizado levando em conta o mapeamento das Inteligências Múltiplas

Fig. 3. Estratégia PICO

Com base nesta estratégia foram propostas três questões específicas que visam buscar subsídios na literatura.

QP1. Quais técnicas são mais utilizadas no ensino de Lógica de Programação: gamificação, robótica ou a linguagem de programação Scratch?

QP2. Quais os resultados obtidos quando estas técnicas são utilizadas no aprendizado de Lógica de Programação?

QP3. As técnicas foram escolhidas baseadas no conhecimento prévio das inteligências múltiplas dos estudantes?

A questão QP1, *quais técnicas são mais utilizadas no ensino de Programação: gamificação, robótica ou a linguagem de programação Scratch*, tem o objetivo de quantificar as técnicas escolhidas pelos docentes para ensinar Programação.

A questão QP2, *quais os resultados obtidos quando estas técnicas são utilizadas no aprendizado de Programação*, tem o objetivo de identificar se as diferentes abordagens de ensino podem facilitar o aprendizado, sem considerar a estimulação das inteligências múltiplas dos alunos.

A questão QP3, *as técnicas foram escolhidas baseadas no conhecimento prévio das inteligências múltiplas dos estudantes*, tem o objetivo de verificar se os docentes têm conhecimento dos tipos de inteligências múltiplas que seus alunos têm mais aflorados, e se este conhecimento está sendo considerado na escolha da metodologia de ensino a ser adotada.

B. Levantamento Inicial dos Artigos

Visando explorar as dificuldades e soluções propostas para o ensino-aprendizagem de Lógica de Programação a nível nacional, as buscas foram no portal de publicações da Comissão Especial de Informática (CEIE) sendo considerados os artigos publicados na Revista Brasileira de Informática na Educação (RBIE), nos anais do Simpósio Brasileiro de Informática na Educação (SBIE), nos anais dos Workshops de Informática na Escola (WIE) e do Congresso Brasileiro de Informática na Educação (CBIE) e na Jornada de atualização em Informática na Escola (JAIE) entre 2012 e 2017. Esta delimitação temporal deve-se ao dinamismo da área em face aos avanços tecnológicos.

Conforme informações publicadas no portal da CEIE [13], a Revista Brasileira de Informática na Educação (RBIE)

foi criada em 1997 e publica trabalhos de excelência realizados por profissionais e pesquisadores na área de Informática em Educação. Todos os trabalhos são avaliados por pelo menos dois avaliadores (*double blind*) mais um membro do corpo editorial da RBIE. A revista é indexada em diversas bases e bibliotecas digitais como: Biblioteca Digital Brasileira de Computação, EBSCO, Google Scholar, Latindex e Portal de Periódicos da CAPES entre outros.

O Simpósio Brasileiro de Informática na Educação (SBIE) é um evento promovido anualmente desde 1990 pela Comissão Especial de Informática na Educação da Sociedade Brasileira de Computação (SBC) e tem como objetivo divulgar a produção científica nacional na área de Informática em Educação e proporcionar um ambiente para a troca de experiências e ideias entre profissionais, estudantes e pesquisadores nacionais e estrangeiros.

O Workshop de Informática na Escola (WIE) é um evento realizado desde 1995 e tem a finalidade de integrar os pesquisadores da Informática na Educação e os professores das escolas. Nos anais dos Workshops do Congresso Brasileiro de Informática na Educação são divulgados os trabalhos dos workshops temáticos realizados no Congresso Brasileiro de Informática na Educação, e na Jornada de Atualização em Informática na Educação (JAIE) é publicado periodicamente o material dos minicursos do CBIE.

A busca no portal de publicações da Comissão Especial de Informática na Educação (CEIE) ocorreu no dia 11 de novembro de 2017 e a definição dos termos e dos conectores lógicos AND ou OR foi baseada em várias tentativas. As *strings*: *lógica de programação AND inteligências múltiplas AND (games OR robótica OR Scratch)*, e *lógica de programação AND inteligências múltiplas OR (games OR robótica OR Scratch)* não retornaram com publicações.

O formato booleano da *string* de busca que retornou com o maior número de artigos pertinentes e que foi utilizada nesta pesquisa é:

lógica de programação OR inteligências múltiplas OR games OR robótica OR Scratch.

A partir da busca na literatura brasileira, retornaram-se 64 artigos, porém como o uso da *string* não garante que eles sejam relevantes em relação às questões desta pesquisa, foi realizada uma pré-análise para identificar os estudos que apresentam as palavras-chave no título ou no *abstract*, o que acabou resultando na seleção de 35 artigos (Tabela I).

TABELA I. PESQUISA NO PORTAL DE PUBLICAÇÕES DA CEIE

ARTIGOS	TOTAL	PRÉ SELECIONADOS
RBIE	8	2
SBIE	18	10
CBIE	14	11
WIE	23	12
JAIE	1	0
TOTAL	64	35

Durante esta fase foi observado o quantitativo de publicação anual sobre o tema no portal de publicações da Comissão Especial de Informática (Fig.4).

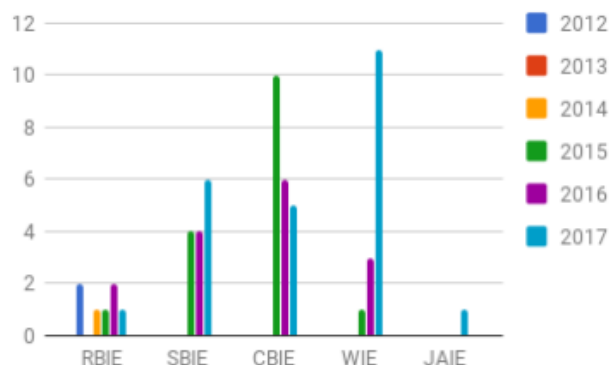


Fig. 4. Publicações no portal da CEIE por ano

Analisando o gráfico, em 2017 foi encontrado nos anais do Workshop de Informática na Escola o maior número de publicações relativas ao tema, o que reflete a importância atual da temática em estudo.

C. Aplicação dos critérios de inclusão e exclusão

Após a eliminação dos artigos que não continham as palavras-chave no título ou no *abstract* procedeu-se a leitura dos *abstracts* dos 35 artigos selecionados. A análise dos artigos seguiu-se por meio dos critérios de inclusão e exclusão definidos no protocolo desta revisão sistemática, levando em consideração o ano de publicação, estratégias para o ensino de Programação, informações sobre o levantamento das Inteligências Múltiplas dos estudantes e que fossem artigos completos (Fig.5).

Critérios de inclusão

- I1- Trabalhos publicados entre 2012 e 2017
- I2- Trabalhos que definam estratégias para o ensino, mesmo que não se refiram às inteligências múltiplas
- I3- Trabalhos que apresentam métodos de estimulação das IM
- I4- Trabalhos que respondam às questões da pesquisa

Critérios de exclusão

- E1- Trabalhos que não foram publicados entre 2012 e 2017
- E2- Trabalhos que não considerem o aprendizado de LP
- E3- Trabalhos publicados como artigos curtos ou poster
- E4- Trabalhos que apresentam resultados, sem o método/técnica utilizados
- E5- Trabalhos duplicados

Fig. 5. Critérios de inclusão e exclusão

III. RESULTADOS OBTIDOS E DISCUSSÃO

A. Análise Quantitativa do Processo de Seleção

A aplicação dos critérios de inclusão e exclusão foi realizada a partir da leitura do resumo e palavras-chaves, levando-se em consideração que foram publicados em periódicos bem avaliados e revisados por pares. Ao final

desta etapa foram excluídos 15 artigos e selecionados 20 (Tabela II).

TABELA II. ARTIGOS SELECIONADOS

	Número de artigos selecionados	Artigos selecionados (Anexo)
WIE	07	[3,5,8,10,12,15,17]
CBIE	06	[1,4,7,13,16,20]
SBIE	07	[2,6,9,11,14,18,19]

B. Estratégia de extração de dados e sumarização das informações

A avaliação da qualidade dos estudos primários é fundamental para validar o processo de inclusão e exclusão, assim como o registro das informações. Na fase de extração de dados foram lidos integralmente os 20 artigos, e para sistematizar o processo foi utilizado um formulário para extração dos dados sintetizados de forma a responder às questões da pesquisa.

Para cada artigo selecionado foram extraídos os seguintes dados: local e ano de publicação, título, autor, hipótese avaliada, público-alvo, tamanho da amostra, métodos utilizados e resultados obtidos.

C. Análise do Processo de Seleção

1) Síntese relacionada à QP1

Em relação à primeira questão de pesquisa (QP1): “*Quais técnicas são mais utilizadas no ensino de Programação: gamificação, robótica ou a linguagem de programação Scratch?*”

Considerando-se as publicações sobre o assunto, a linguagem de programação *Scratch* foi a técnica mais utilizada no Ensino Fundamental, robótica no Médio e gamificação no Ensino Superior (Fig.6).

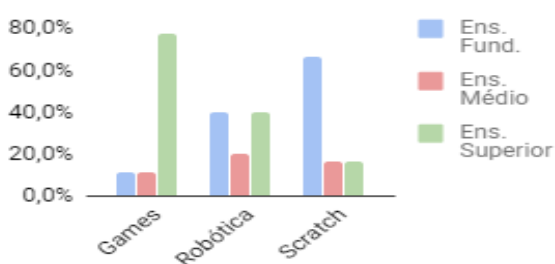


Fig. 6. Técnicas mais utilizadas no ensino de LP

2) Síntese relacionada à QP2

Em relação à segunda questão de pesquisa (QP2): “*Quais os resultados obtidos quando estas técnicas são utilizadas no aprendizado de Programação?*”

A linguagem de programação *Scratch* [14], inspirada nas ideias de Logo, foi criada pelo Media Lab do Instituto de Tecnologia de Massachussetts (MIT) para crianças a partir dos sete anos e possibilitou uma aproximação entre os conteúdos curriculares da disciplina Matemática com as tecnologias digitais, auxiliando o aprendizado de conceitos matemáticos e computacionais.

Nos trabalhos considerados nesta pesquisa, a *Scratch* foi a mais utilizada no Ensino Fundamental (66,6%) e em todos os casos foram observados ganhos reais no aprendizado de Programação (Tabela III).

TABELA III. RESULTADOS NO ENSINO FUNDAMENTAL

	AUTOR	OBJETIVO	RESULTADOS
1	Duarte, K.T.N.; Silveira, T.R.S.; Borges, M.A.F. (2017)	Ensinar LP através da ferramenta <i>Scratch 2.0</i>	Oficina de aprendizado interativo e lúdico permitiu a aprendizado do aluno
2	Ribeiro, S.S.; Melo, A.M (2017)	Desenvolver o pensamento computacional e lógico utilizando o ambiente <i>Scratch</i>	Desenvolvimento de software por crianças, do raciocínio lógico e do pensamento computacional
3	Silva, G.T.; Souza, J.L.; Silva, L.A.M. (2016)	Auxiliar o aprendizado de LP através da ferramenta <i>Scratch</i>	Trabalho no laboratório permitiu aprendizado mais efetivo
4	Silva, J.L.; Gens, C.; Santos, C.P. (2017)	Auxiliar o aprendizado de LP através do <i>jogo Play Code Dog</i>	Ferramenta propiciou flexibilidade e criatividade aos alunos, mas há necessidade de capacitar os educadores
5	Aguiar, Y.Q.; Maciel, B. K.; Mattos, S.D.G.; Soares, L.B.; Oliveira, V.M. (2015)	Estimular o aprendizado de LP através de robótica : Kit Educativo LEGO <i>Mindstorms</i>	Curso instigou nos alunos uma nova forma de pensar auxiliando na Lógica de Programação e em avaliações que exigem pensamento lógico e abstrato
6	Martins, L.A.S.; Brelaz, A.S.; Nascimento, G.R.; Alfaia, R.M.; Martins, T.S. (2016)	Ensinar LP aplicada à robótica através de recursos lúdicos como jogos online, montagem de robôs e programação em blocos: Aplicativo <i>Scratch</i> , plataforma LEGO	Grande capacidade de abstração e resolução dos programas, dificuldade para entender os conceitos.

No Ensino Médio observou-se uma discreta tendência no uso da robótica como ferramenta para auxiliar o estudante a construir uma lógica de raciocínio de forma simples, porém todas as ferramentas aplicadas contribuíram para a maior eficácia no aprendizado de Programação, sejam games, robótica ou *Scratch* (Tabela IV).

No Ensino Superior, o desenvolvimento de ambientes gamificados foi eficiente como facilitador do aprendizado de Lógica de Programação, motivando os estudantes e permitindo o desenvolvimento do raciocínio lógico, da prática na codificação e da montagem de algoritmos (Tabela V).

TABELA IV. RESULTADOS NO ENSINO MÉDIO

	AUTOR	OBJETIVO	RESULTADOS
1	Leôncio, L.N.; Sousa, C.C.; Sousa, R.P. (2017)	Usar o MIT <i>AppInventor</i> no ensino de LP	A interface e a plataforma Android facilitam a prática da LP
2	Cambruzzi, E.; Souza, R.M. (2015)	Estimular o aprendizado de LP construindo, aplicando e avaliando Objetos de Aprendizagem baseados na robótica educativa	Contribuição para a aprendizagem de forma lúdica e interativa
3	Zanchett, G.A.; Vahldick, A.; Raabe.A. (2015)	Abordar LP a partir dos jogos de programar: <i>Code Combat, Light Bot e No Bug's Snack Bar</i>	Os jogos <i>Code Combat e Light Bot</i> se mostraram eficientes como facilitadores do aprendizado de LP
4	Oliveira, M.V.; Rodrigues, L.C.; Queiroga, A.P.G. (2016)	Auxiliar o aprendizado de LP através da ferramenta Scratch	Aprendizado facilitado de LP gerando confiança e desenvolvendo a criatividade

TABELA V. RESULTADOS NO ENSINO SUPERIOR

	AUTOR	OBJETIVO	RESULTADOS
1	Roberti JR, W.C.; Lavorato, A.S.; Lucarelli, D.C.; Machado, A.F.V. (2017)	Ensinar Lógica de Programação através do jogo <i>Defense of the Ancients 2</i>	Desenvolveu o interesse e auxiliou para firmar conceitos no aprendizado de Lógica de Programação
2	Moissa, B.; Santos, L.V.; Klock, A.C.T.; Gasparini, I.; Koczinski (2016)	Aplicar o modelo ADDIE em material para ensinar Lógica de Programação	Curso bem avaliado e facilitador do aprendizado do aluno
3	Rhenns, C.L.; Raabe, A.L.A.; Santana, A. (2016)	Trabalhar os conceitos de matemática preliminar e raciocínio lógico através do jogo <i>Ninja Prog</i>	Melhora na resolução de problemas de raciocínio lógico e matemática através do jogo de ação
4	Gomes, M.S.; Amaral, E.M.H. (2016)	Utilizar ferramenta para simplificar a depuração de códigos em C	Dificuldade dos alunos amenizada com a ferramenta C fácil
5	Carvalho, R.; Rosa, P.; Machado, J.V.; Ribeiro Jr, J.G.; Costa, G. (2015)	Utilizar um aplicativo e um <i>hardware</i> para facilitar o aprendizado de uma lógica de raciocínio	Ferramenta facilitou o aprendizado de Lógica de Programação
6	Galdino, C.B.T.; Silva Neto, S.R.; Costa, E.B. (2015)	Ensinar Lógica de Programação através do jogo <i>Kid Coder</i>	Ferramenta permitiu praticar codificação e montagem de algoritmos

7	Nagano, L. (2016)	Ensinar Lógica de Programação baseado na indução-dedução através de exemplos	Método facilita a aprendizagem de Lógica de Programação
8	Souza, M.S.C.; Costa, F.A.M.; Silva, V.L.; Terra, D.C. (2016)	Utilizar o jogo <i>Lord of Code (LoC)</i>	Ferramenta auxilia na sintaxe de programação
9	Barbosa, B.; Silva,S.; Sousa, B. (2017)	Ensinar Lógica de Programação através do jogo eletrônico Tri-Logic	Ambiente de apoio facilitador aliando jogo com técnicas de mineração de dados
10	Tabuti, L.M.; Nakamura, R. (2015)	Coletar dados sobre os jogos digitais de Lógica	Estudos que definem estes métodos necessitam ser mais explorados

3) Síntese relacionada à QP3

Em relação à terceira questão de pesquisa (QP3): “As técnicas foram escolhidas baseadas no conhecimento prévio das inteligências múltiplas dos estudantes?”

A partir da busca realizada não foi encontrado nenhum trabalho que discutisse ou levantasse a temática da necessidade do conhecimento prévio das Inteligências Múltiplas dos estudantes como ponto de partida para definir qual a melhor técnica de ensino de programação a ser implementada à sua clientela.

IV. CONSIDERAÇÕES FINAIS

A partir da revisão sistemática é possível ter uma visão do estado da arte das técnicas que visam a melhoria do ensino-aprendizagem de Programação. Analisando o material extraído é possível obter informações importantes sobre as ferramentas que são utilizadas, sendo possível constatar que os conceitos de lógica são inseridos a partir do Ensino Fundamental I, porém a maioria das pesquisas foi desenvolvida no ensino superior, conforme mostra a Fig. 7.

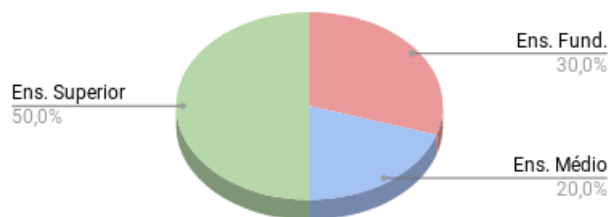


Fig. 7. Público alvo das publicações

Os resultados dos artigos foram demonstrados a partir da observação dos grupos focais e testes de avaliação. Quanto às estratégias utilizadas como facilitadoras da aprendizagem de Programação há uma tendência no uso de games e a ferramenta *Scratch* (Fig. 8).

Os *games* são ferramentas que aumentam a motivação e o interesse dos estudantes, e há indícios de melhora no

desempenho e redução na evasão, quando usados no apoio ao ensino da programação no ensino superior.

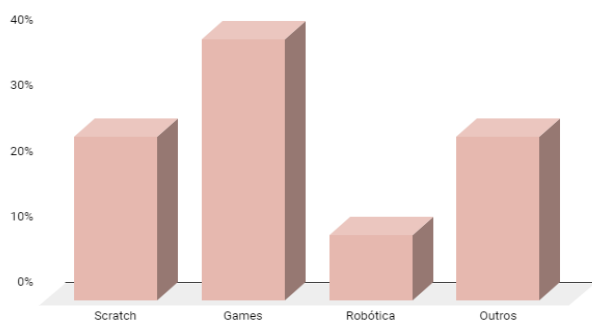


Fig. 8. Estratégias utilizadas no ensino de Programação

A ferramenta *Scratch* pode ser aplicada em diferentes níveis de escolaridade e propicia que os conteúdos produzidos sejam socializados.

Os trabalhos demonstram que há um interesse crescente em iniciar o ensino de programação a partir do ensino fundamental e que novas metodologias, técnicas e ferramentas tecnológicas têm sido desenvolvidas e testadas por docentes e pesquisadores.

Com o mapeamento das técnicas utilizadas no processo de ensino-aprendizagem da disciplina de Programação foi verificado que todas apresentam potencial para influenciar positivamente o desempenho dos estudantes, porém não foi encontrada nenhuma evidência sobre o prévio mapeamento das inteligências múltiplas dos grupos focais em relação às estratégias de ensino escolhidas.

Considerando que cada pessoa tem um modo particular de se apropriar do conhecimento, para potencializar os resultados do processo de ensino-aprendizagem é necessário investigar todas as competências intelectuais ou inteligências nos estudantes, como elas interagem ou podem ser estimuladas e propor atividades que estimulem as mais desenvolvidas [9].

Ao considerar as inteligências múltiplas é possível reconhecer diferentes habilidades e talentos dos estudantes e viabilizar rotas de aprendizagem personalizadas [18].

Como trabalho futuro pretende-se investigar o desenvolvimento de mecanismos apropriados de inovação tecnológica que facilitem o aprendizado dos estudantes com maior dificuldade em programação considerando a existência de Inteligências Múltiplas. Desta forma, pretende-se que uma real construção do conhecimento e do desenvolvimento de técnicas variadas a partir do conhecimento das Inteligências Múltiplas possibilitem melhores resultados no ensino da programação.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] S.S. Prietchand T.A. Pazeto, "Estudo sobre a Evasão em um Curso de Licenciatura em Informática e Considerações para Melhorias". Anais do VIII Workshop de Educação e Informática Bahia-Alagoas-Sergipe, WEIBASE 2010, Maceió, AL.

[2] J. C. R. P. Júnior and C. E. Rapkiewicz, "O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura". III Workshop de Educação em Computação e Informática do estado de Minas Gerais, WEIMIG 2004, Belo Horizonte, MG.

[3] A. L. A. Raabeand J. M. C. Silva, "Um Ambiente para Atendimento as Dificuldades de Aprendizagem de Algoritmos". XXV Congresso da Sociedade Brasileira de Computação, 2005, São Leopoldo, RS.

[4] M. Berry, "Computing in the national curriculum a guide for primary teacher". Disponível em: <http://www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf>. [Acessado em 04/08/2018].

[5] Plataforma Code.org. Disponível em: <http://br.code.org>. [Acessado em 04/08/2018].

[6] Madcode. Disponível em: <http://madecode.com.br>. [Acessado em 04/08/2018].

[7] Let'sCodeAcademy. Disponível em: <http://letscode-academy.com>. [Acessado em 04/08/2018].

[8] L.D.O. Maia et. al., "A Robótica como Ambiente de Programação Utilizando o Kit Lego Mindstorms". Anais do XVII Simpósio Brasileiro de Informática na Educação, SBIE 2006, Brasília, DF.

[9] H. Gardner, "Inteligência: um conceito reformulado: o criador da teoria de inteligências múltiplas explica e expande suas idéias com o enfoque no século XXI, Rio de Janeiro: Objetiva, 2001.

[10] S. A. Walter, M. A. Schneider, J. R. Frega, M. J. C. S. Domingues, "Similaridades e divergências no desenvolvimento das inteligências múltiplas de um Curso de Ciências Contábeis: um comparativo entre cursos, turmas e gêneros". Revista Brasileira de Gestão de Negócios, v. 11, n. 31, 2009, pp. 134-151.

[11] B. Kitchenham, "Procedures for Performing Systematic Reviews". Joint Technical Report, TR/SE-0401 and NICTA 0400011T.1, Keele University. Disponível em: http://www.idi.ntnu.no/emner/empse/papers/kitchenham_2004.pdf. [Acessado em 04/08/2018].

[12] C.M.C. Santos, C.A.M. Pimenta and M.R.C. Nobre, "The PICO strategy for the research question construction and evidence search," *Rev. Latino-Am. Enfermagem* [online]. 2007, vol.15, n.3, pp. 508-511. Disponível em: <http://dx.doi.org/10.1590/S0104-11692007000300023>. [Acessado em 04/08/2018].

[13] Portal de Publicações da CEIE. Disponível em: www.br-ie.org/pub/index.php/index. [Acessado em 04/08/2018].

[14] Scratch. Disponível em: <http://www.scratchbrasil.net.br/index.php/sobre-o-scratch.html>. [Acessado em 04/08/2018].

[15] J. Biolchini, P.G. Mian, A.C.C. Natalli and G.H. Travassos, "Systematic Review in Software Engineering". Technical Report RT-ES 679/05, 2005, COPPE/UFRJ. Disponível em: <http://alarcos.inf-cr.uclm.es/doc/MetoTecInf/Articulos/es67905.pdf>. [Acessado em 04/08/2018].

[16] M.C. De-La-Torre-Ugarte-Guanilo, R.F. Takahashi and M.R. Bertolozzi, "Revisão sistemática: noções gerais". *Rev. esc. enferm. USP* [online]. 2011, vol.45, n.5, pp.1260-1266. Disponível em: <http://dx.doi.org/10.1590/S0080-62342011000500033>. [Acessado em 04/08/2018].

[17] T. Dyba, T. Dingsoyr, G.K. Hanssen, "Applying Systematic Reviews to Diverse Study Types: An Experience Report". First International Symposium on Empirical Software Engineering and Measurement, IEEE, 2007. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4343750>. [Acessado em 04/08/2018].

[18] Z.Cataldi, "Modelo delalumno", Revista Iberoamericana de Tecnología enEducación y Educaciónen Tecnología, pp. 30-32, 2009.

ANEXO: ARTIGOS CONSIDERADOS NA REVISÃO SISTEMÁTICA

1. Aguiar, Y.Q.; Maciel, B. K.; Mattos, S.D.G.; Soares, L.B.; Oliveira, V.M. (2015). Introdução à Robótica e Estímulo à Lógica de Programação no Ensino Básico Utilizando o Kit Educativo LEGO® Mindstorms. Anais dos Workshops do CBIE 2015. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2015.1418>
2. Barbosa, B.; Silva,S.; Sousa, B. (2017). Tri-Logic Proposta Lúdica Gamificada para o Ensino e Aprendizagem da Lógica de Programação com o Uso da Mineração de Dados como Ferramenta de Auxílio ao Professor. Anais do SBIE 2017. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2017.1754>
3. Cambuzzi, E.; Souza, R.M. (2015). Robótica Educativa na aprendizagem de Lógica de Programação: Aplicação e análise. Anais do WIE 2015. DOI: <http://dx.doi.org/10.5753/cbie.wie.2015.21>
4. Carvalho, R.; Rosa, P.; Machado, J.V.; Ribeiro Jr, J.G.; Costa, G. (2015). Ferramenta para Auxílio na Aprendizagem de Lógica de

Programação em Sistemas Informatizados. Anais dos Workshops do CBIE 2015. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2015.1144>

5. Duarte, K.T.N.; Silveira, T.R.S.; Borges, M.A.F. (2017). Abordagem para o Ensino da Lógica de Programação em Escolas do Ensino Fundamental II através da Ferramenta Scratch 2.0. Anais do WIE 2017. DOI: <http://dx.doi.org/10.5753/cbie.wie.2017.175>

6. Galdino, C.B.T.; Silva Neto, S.R.; Costa, E.B. (2015). KidCoder: Uma Proposta de Ensino de Programação de forma Lúdica. Anais do SBIE 2015. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2015.687>

7. Gomes, M.S.; Amaral, E.M.H. (2016). Uma Proposta de Ferramenta para Simplificar a Depuração de Códigos em C, por Alunos Iniciantes. Anais dos Workshops do CBIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2016.1029>

8. Leôncio, L.N.; Sousa, C.C.; Sousa, R.P. (2017). Programação em blocos com o MIT App Inventor: Um relato de experiência com alunos do ensino médio. Anais do WIE 2017. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2017.1159>

9. Martins, L.A.S. Ensinando Lógica de Programação aplicada a Robótica para alunos do Ensino Fundamental. Anais do SBIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2016.31>

10. Moissa, B.; Santos, L.V.; Klock, A.C.T.; Gasparini, I.; Komczinski (2016). Um minicurso online de Algoritmos como apoio às disciplinas iniciais da graduação: preparação, execução e resultados sobre a satisfação dos alunos. Anais do WIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.wie.2016.466>

11. Nagano, L. (2016). Ensino de Lógica de Programação baseado na indução-dedução através de exemplos. Anais do SBIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2016.1352>

12. Oliveira, M.V.; Rodrigues, L.C.; Queiroga, A.P.G. (2016). Material didático lúdico: uso da ferramenta Scratch para auxílio no aprendizado de lógica da programação. Anais do WIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.wie.2016.359>

13. Rhenns, C.L.; Raabe, A.L.A.; Santana, A. (2016). Ninja Prog - Jogo de conceitos matemáticos e lógica de programação. Anais dos Workshops do CBIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2016.159>

14. Ribeiro, S.S.; Melo, A.M. (2017). Um Método para o Desenvolvimento de Software com Crianças Utilizando o Ambiente Scratch. Anais do SBIE 2017. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2017.1027>

15. Roberti Jr, W.C.; Lavorato, A.S.; Lucarelli, D.C.; Machado, A.F.V. (2017). Ensino de Lógica de Programação através do Jogo Defense of the Ancients 2. Anais do WIE 2017. DOI: <http://dx.doi.org/10.5753/cbie.wie.2017.648>

16. Silva, G.T.; Souza, J.L.; Silva, L.A.M. (2016). Aplicação da Ferramenta Scratch para o Aprendizado de Programação no Ensino Fundamental I. Anais dos Workshops do CBIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2016.1285>

17. Silva, J.L.; Gens, C.; Santos, C.P. (2017). Lógica de Programação: Iniciação Lúdica com Play Code Dog. Anais do WIE 2017. DOI: <http://dx.doi.org/10.5753/cbie.wie.2017.108>

18. Souza, M.S.C.; Costa, F.A.M.; Silva, V.L.; Terra, D.C. (2016). Lord of Code: Uma Ferramenta de Apoio ao Ensino da Programação. Anais do SBIE 2016. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2016.1316>

19. Tabuti, L.M.; Nakamura, R. (2015). Métodos para o Desenvolvimento de Jogos Digitais de Lógica: Uma Revisão Sistemática. Anais do SBIE 2015. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2015.41>

20. Zanchett, G.A.; Vahldick, A.; Raabe, A. (2015). Jogos de Programar como uma Abordagem para os Primeiros Contatos dos Estudantes com a Programação. Anais dos Workshops do CBIE 2015. DOI: <http://dx.doi.org/10.5753/cbie.wcbie.2015.1485>