



Emanuel Leite Oliveira da Silva

**Uma Análise do Impacto da Experiência Prévia  
com Pensamento Computacional no  
Desempenho de Estudantes em Programação  
no Ensino Superior**

Recife

2019

Emanuel Leite Oliveira da Silva

# **Uma Análise do Impacto da Experiência Prévia com Pensamento Computacional no Desempenho de Estudantes em Programação no Ensino Superior**

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Universidade Federal Rural de Pernambuco – UFRPE  
Departamento de Computação  
Curso de Bacharelado em Ciência da Computação

Orientadora: Profa. Dra. Taciana Pontual Falcão

Recife  
2019

Dados Internacionais de Catalogação na Publicação  
Universidade Federal Rural de Pernambuco  
Sistema Integrado de Bibliotecas  
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

---

S586a Silva, Emanuel Leite Oliveira da  
Uma Análise do Impacto da Experiência Prévia com Pensamento Computacional no Desempenho de  
Estudantes em Programação no Ensino Superior / Emanuel Leite Oliveira da Silva. - 2019.  
48 f.

Orientadora: Taciana Pontual da Rocha Falcao.  
Inclui referências e apêndice(s).

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal Rural de Pernambuco,  
Bacharelado em Ciência da Computação, Recife, 2019.

1. Pensamento computacional. 2. Graduação. 3. Licenciatura em computação. 4. Raciocínio lógico. 5.  
Pensamento algorítmico. I. Falcao, Taciana Pontual da Rocha, orient. II. Título

CDD 004

---



MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO  
UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO (UFRPE)  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

<http://www.bcc.ufrpe.br>

**FICHA DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO**

Trabalho defendido por **EMANUEL LEITE OLIVEIRA DA SILVA** às 16:30 do dia 06 de dezembro de 2019, no Auditório do Departamento de Computação - DC – Sala 07, como requisito para conclusão do curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, intitulado " **Uma Análise do Impacto da Experiência Prévia com Pensamento Computacional no Desempenho de Estudantes em Programação no Ensino Superior** ", orientado por Taciana Pontual da Rocha Falcão e aprovado pela seguinte banca examinadora:

---

Taciana Pontual da Rocha Falcão  
DC/UFRPE

---

Rozelma Soares de França  
DC/UFRPE

# Resumo

Este trabalho se propôs a estudar o efeito do contato prévio com o Pensamento Computacional em alunos de cursos superiores em computação. O Pensamento Computacional é uma habilidade que visa o desenvolvimento do raciocínio lógico e pensamento algorítmico de uma forma contínua e por toda a vida do indivíduo, auxiliando-o para solucionar problemas da vida pessoal e profissional utilizando as técnicas da ciência da computação. Segundo pesquisas, mais de 50% dos alunos dos cursos de computação irão evadir o curso e um dos principais motivos é a dificuldade em aprender e assimilar os conceitos básicos e avançados da programação, ficando desmotivados. Com isso, este trabalho investigou a viabilidade do uso do pensamento computacional para ajudar esses alunos com dificuldades no aprendizado de programação. Portanto, foram identificados dois perfis de alunos, que tiveram contato com o Pensamento Computacional antes e depois de cursarem Programação, e aplicados questionários para avaliar as perspectivas que eles tiveram sobre a disciplina e o seu benefício, se foi produtivo ou não o uso do Pensamento Computacional. Também foram entrevistados dois professores do curso de Licenciatura em Computação da UFRPE para examinar a perspectiva deles em relação ao Pensamento Computacional no desempenho dos alunos, comparando os alunos que tiveram o contato antes e depois de cursarem Programação. Sob a perspectiva dos alunos o uso do Pensamento Computacional os auxiliam no desenvolvimento cognitivo, melhorando o raciocínio lógico e o pensamento algorítmico, e no aprendizado de Programação. Os professores creem que o Pensamento Computacional prepara cognitivamente os alunos para Programação, reduzindo o esforço em reaprender os conceitos básicos e que veem essa abordagem como um aprimoramento para os alunos.

**Palavras-chave:** pensamento computacional, graduação, licenciatura em computação, raciocínio lógico, pensamento algorítmico.

# Abstract

This paper aims to study the effect of previous contact with Computational Thinking in students of higher education courses. Computational Thinking is a skill that aims to develop logical thinking and algorithmic thinking on an ongoing and lifelong basis, helping them to solve personal and professional life problems using the techniques of computer science. According to research, more than 50% of students in computer courses will drop out of the course and one of the main reasons is the difficulty in learning and assimilating the basic and advanced concepts of programming, becoming unmotivated. Thus, this work investigated the feasibility of using computational thinking to help those students with programming learning difficulties. Therefore, two student profiles were identified, who had contact with Computational Thinking before and after attending Programming, and questionnaires were applied to evaluate the perspectives they had on the discipline and its benefit, whether the use of Computational Thinking was productive or not. Two teachers from the UFRPE Computer Degree course were also interviewed to examine their perspective on Computational Thinking on student performance, comparing students who had contact before and after attending Programming. From the students' perspective, the use of Computational Thinking assists them in cognitive development, improving logical thinking and algorithmic thinking, and programming learning. Teachers believe that Computational Thinking cognitively prepares students for Programming, reducing the effort to assimilate the basics and seeing this approach as an improvement for students.

**Keywords:** computational thinking, graduation, computer degree, logical reasoning, algorithmic thinking.

# Lista de tabelas

Tabela 1 – Resumo dos trabalhos relacionados . . . . .	20
Tabela 2 – Distribuição dos participantes AP1 por IES . . . . .	24
Tabela 3 – Motivos para alunos AP1 cursarem Licenciatura em Computação . . . . .	25
Tabela 4 – Níveis de raciocínio lógico dos alunos AP1 . . . . .	26
Tabela 5 – Níveis de pensamento algorítmico dos alunos AP1 . . . . .	26
Tabela 6 – Distribuição dos participantes AP2 por IES . . . . .	27
Tabela 7 – Motivos para alunos AP2 cursar Licenciatura em Computação . . . . .	28
Tabela 8 – Motivos para evasão dos alunos AP2 . . . . .	29
Tabela 9 – Níveis de raciocínio lógico dos alunos AP2 . . . . .	30
Tabela 10 – Níveis de pensamento algorítmico dos alunos AP2 . . . . .	30

# Lista de abreviaturas e siglas

AP1	Aluno do Perfil 1
AP2	Aluno do Perfil 2
IFB	Instituto Federal de Brasília
IFBA	Instituto Federal da Bahia
IFFar	Instituto Federal Farroupilha
IES	Instituições de Ensino Superior
IFMG	Instituto Federal de Minas Gerais
IFRN	Instituto Federal do Rio Grande do Norte
IFTM	Instituto Federal do Triângulo Mineiro
IFTO	Instituto Federal do Tocantis
LC	Licenciatura em Computação
MEC	Ministério da Educação
PC	Pensamento Computacional
RFCGC	Referenciais de Formação dos Cursos de Graduação em Computação
RSL	Revisão Sistemática da Literatura
SBC	Sociedade Brasileira de Computação
TR	Tipo de Relação
UFBA	Universidade Federal da Bahia
UFERSA	Universidade Federal Rural do Semi-Árido
UFJF	Universidade Federal de Juiz de Fora
UFPB	Universidade Federal da Paraíba
UFPR	Universidade Federal do Paraná
UFRA	Universidade Federal Rural da Amazônia



UFRPE	Universidade Federal Rural de Pernambuco
UFSB	Universidade Federal do Sul da Bahia
UTFPR	Universidade Tecnológica Federal do Paraná
VA	Verificação de Aprendizagem

# Sumário

1	INTRODUÇÃO	8
1.1	Problema de pesquisa	9
1.2	Objetivos	9
2	PENSAMENTO COMPUTACIONAL	11
3	DIFICULDADES NO APRENDIZADO EM PROGRAMAÇÃO NO ENSINO SUPERIOR	14
3.1	Problemas identificados pelos professores	15
3.2	Problemas identificados pelos alunos	15
3.3	Oportunidades	16
4	TRABALHOS RELACIONADOS	17
5	METODOLOGIA	21
6	RESULTADOS	24
6.1	Entrevistas com alunos do perfil AP1	24
6.2	Entrevistas com alunos do perfil AP2	27
6.3	Entrevistas com professores	31
7	CONCLUSÕES E TRABALHOS FUTUROS	34
7.1	Trabalhos futuros	36
	REFERÊNCIAS	38
A	QUESTIONÁRIO DOS ALUNOS AP1	41
B	QUESTIONÁRIO DOS ALUNOS AP2	44

# 1 Introdução

O Ministério da Educação (MEC) classifica a evasão na educação superior brasileira como a não conclusão e afastamento permanente da graduação (MEC, 1996). Cursos superiores brasileiros na área de Computação possuem em média um índice de evasão de 75%, e as disciplinas que envolvem algoritmos estão entre os principais fatores para a evasão do curso (HOED, 2016). O estudo de Hoed (2016) mostra estratégias usadas por algumas universidades, que impactaram na evasão e retenção, podendo ser replicadas aos cursos de computação para reduzir as dificuldades que os alunos têm com disciplinas que envolvam a resolução de problemas. Uma abordagem apoiada pelo estudo é a de que professores utilizem ferramentas educacionais que permitam que os alunos adquiram a capacidade de pensar computacionalmente antes de aprenderem a escrever e desenvolver programas.

O Pensamento Computacional (PC) é uma habilidade que vem ganhando atenção em pesquisas para influenciar na redução da evasão em cursos de Computação. O PC consiste em utilizar as técnicas da ciência da computação para estimular a criatividade e o raciocínio lógico ao lidar com os problemas com que nos deparamos em situações profissionais e pessoais, e em qualquer domínio do conhecimento (WING, 2006). Olhar a ciência da computação apenas como uma área focada em programar máquinas é desconsiderar todo o engajamento e conhecimento criado para encontrar soluções aos desafios da humanidade, o que muitas vezes exige encontrar novas formas de superá-los (WING, 2006).

O papel de um cientista da computação inclui, portanto, disseminar e provocar a conscientização de que o PC é uma habilidade para o avanço não só da ciência, mas da sociedade (WING, 2006), trazendo ao público e pesquisadores um novo mecanismo para o desenvolvimento cognitivo, corroborando que essa habilidade deve ser mantida durante toda a vida. Entretanto, nem todos os cursos superiores de computação oferecem uma abordagem de ensino que aproveite o PC e ajude na superação das dificuldades em disciplinas que o envolvem (HOED, 2016).

Araújo, Silveira e Mattos (2018) demonstram que o atraso tecnológico do Brasil já é uma problemática reconhecida há alguns anos, identificando cinco grandes desafios para a Ciência da Computação brasileira que deveriam ter sido atacados na década de 2006 a 2016, colocando o PC como um tópico em destaque. Trazendo esse contexto para os cursos de graduação, nos Referenciais de Formação para os Cursos de Graduação em Computação (RFCGC), da Sociedade Brasileira de Computação (SBC), o PC é uma das competências desejadas dentro do eixo de fundamentos da

computação nos cursos de Licenciatura em Computação (LC) (ZORZO et al., 2017). Assim, as Instituições de Ensino Superior (IES) deveriam desenvolver estudos e estratégias para encontrar formas de incluir o desenvolvimento dessa habilidade em suas matrizes curriculares, seja introduzindo uma disciplina própria, como já está sendo aplicado em algumas universidades, por projetos de ensino (SCAICO et al., 2012); (VAHLICK et al., 2016); ou com professores associando o PC à didática de suas disciplinas no início do curso quando possível (PONTUAL FALCÃO et al., 2018).

Este trabalho está estruturado em sete capítulos: no Capítulo 1 esta a introdução e contextualização deste trabalho. O Capítulo 2 aborda e explica os princípios do Pensamento Computacional, sua origem e popularização, e o seu papel na ciência da computação. O Capítulo 3 discorre sobre as dificuldades encontradas por alunos e professores quanto ao aprendizado de programação nos cursos de graduação em computação, e também expõe a oportunidade de estudo encontrada que este trabalho aborda. No Capítulo 4 são encontrados os trabalhos relacionados que sustentam o tema de estudo deste. O Capítulo 5 explica a metodologia de pesquisa utilizada para a conclusão deste trabalho. No Capítulo 6 são encontrados os resultados obtidos por este trabalho e no Capítulo 7 as conclusões, discussões e trabalhos futuros a partir dos resultados da pesquisa deste trabalho.

## 1.1 Problema de pesquisa

Considerando-se a necessidade das IES encontrarem mecanismos para reduzir a evasão e retenção em cursos de Computação, o problema a ser investigado na presente pesquisa é: como uma experiência prévia com o pensamento computacional pode influenciar no aprendizado e desempenho dos alunos de programação? Para fazer essa análise, pretende-se considerar não somente as percepções dos alunos e professores, mas também a influência de questões afetivas e aspectos sócio-culturais dos alunos.

## 1.2 Objetivos

Este trabalho tem seus objetivos descritos abaixo.

- Objetivo Geral:
  - Investigar o impacto do contato prévio com pensamento computacional no aprendizado de programação nos alunos dos cursos superiores de computação.
- Objetivos Específicos:

- Examinar a percepção do aprendizado dos estudantes que tiveram contato com PC antes e depois de cursarem programação;
- Examinar a percepção dos professores de programação em relação ao desempenho dos alunos que previamente cursaram PC;
- Identificar pontos de melhoria na disciplina de PC para favorecer o aprendizado de programação;
- Investigar as questões afetivas que os alunos desenvolvem com a disciplina de programação, comparando as visões dos que tiveram e dos que não tiveram contato com PC antes de cursar programação.

## 2 Pensamento Computacional

Seja na vida pessoal ou profissional, o contexto de vida do ser humano é uma sucessão de tentativas de solucionar problemas e a capacidade de raciocínio permitiu que a humanidade fosse adaptando e utilizando o que tinha ao alcance para superar seus obstáculos. Quando desejamos construir uma casa, primeiro temos que preparar o terreno, em seguida fazer os alicerces e então começar a construção das paredes fundamentais e, por último, o teto; um exemplo bem simples e intuitivo do conceito de dividir para conquistar, baseado em todo um raciocínio e um conhecimento por trás dessa decisão. Essa é a essência do pensamento computacional, tentar encontrar meios mais simples e mecanismos melhores para usar, ou criar, soluções para os problemas, usando a abstração, generalização, decomposição, pensamento algorítmico e avaliação (WING, 2006). O termo Pensamento Computacional se popularizou através do artigo de Wing (2006), mas já é discutido por Papert (1980).

Para isso o pensamento computacional visa o despertar de alguns conceitos no dia-a-dia das pessoas, para que consigam enxergar as situações com as quais se deparam de forma lógica, conseguindo analisá-las, reconhecer a grandeza dos problemas e, se necessário, dividindo-os em problemas de menor tamanho, para os quais já pode haver uma solução conhecida (WING, 2006); (CSIZMADIA et al., 2015).

Ao generalizar um problema é possível identificar padrões, reconhecer quando ele pode ser dividido de uma forma que já exista uma solução para um subproblema, ou adaptar as que já existem, e assim encontrar uma solução mais rápida e eficaz (CSIZMADIA et al., 2015). Fazendo uma analogia ao conceito de dividir para conquistar, utilizado por muitos algoritmos de ordenação, o aluno pode visualizar um desafio, como por exemplo, ordenar uma lista de números e propor estratégias de como fazê-lo.

Ao fazer isso o aluno deve conseguir abstrair quaisquer informações não essenciais para que qualquer pessoa consiga utilizar a sua solução, ou algoritmo, de forma objetiva e clara. Informações que variam ou vão além do escopo principal daquela solução são ignoradas após aplicar a abstração do problema, que é outra competência oriunda do pensamento computacional (CSIZMADIA et al., 2015). Isso pode permitir que o aluno consiga estruturar suas ideias de forma mais eficiente e inteligível, sendo um forte indício da compreensão e domínio sobre o que está sendo estudado.

Quando pensamos de forma algorítmica devemos descrever uma solução em forma de etapas, um passo a passo, de forma ordenada e reproduzível, depois de abstrair, reconhecer os padrões e contextos relacionados, e dividir todo o problema, para que qualquer outra pessoa consiga encontrar uma solução ao reproduzir os passos

(CSIZMADIA et al., 2015). Assim o aluno pode criar etapas, de tamanhos diversos e de acordo com o seu próprio conhecimento e capacidades, de forma sequencial e iterativa para desenvolver uma solução aos problemas propostos nas atividades passadas pelo professor.

Após propor uma solução, é necessária uma avaliação se todos os requisitos para resolver o problema inicial foram atendidos, se não há algo que ficou fora de escopo e não foi incluído durante o desenvolvimento, validar se o que foi produzido realmente atende a necessidade de todos envolvidos. É questionar sobre o quão difícil é um problema e qual a forma eficiente e eficaz de resolvê-lo, prevendo e criando mecanismos de recuperação em caso de erros (CSIZMADIA et al., 2015); (WING, 2006). Com uma solução pronta, o aluno pode avaliar se essa solução realmente soluciona o problema exposto e encontrar possíveis pontos de melhoria, verificando a eficiência e a eficácia do próprio resultado.

Desde a publicação de Wing (2006), o estudo sobre o PC vêm ampliando e novos conceitos são adicionados aos já conhecidos, como: organizar, analisar e representar os dados de forma lógica e através de abstrações; automatizar soluções através do pensamento algorítmico; formalizar problemas de modo que sejam computacionalmente possíveis; considerar eficiência e eficácia ao propor soluções a problemas levando em consideração os recursos disponíveis (ISTE, 2011). Além de desenvolver habilidades como: capacidade de lidar com problemas em aberto, se organizar em equipe, lidar com problemas complexos e persistência (ISTE, 2011).

Os professores são responsáveis por difundir esses métodos da ciência da computação em sala de aula, não apenas vendo como etapas do desenvolvimento de um *software*, mas como uma expansão cognitiva dos seus alunos (WING, 2006); (CSIZMADIA et al., 2015).

A ciência da computação não se baseia apenas em escrever programas, sendo responsável por várias técnicas que otimizam a forma com que as pessoas agem em equipe, trabalhando assim o relacionamento interpessoal dos indivíduos, encontrar padrões em problemas e saber como decompor para que uma atividade grande e difícil de ser realizada, se torne várias atividades pequenas e mais simples. É antecipar situações, planejar e aprender (WING, 2006).

Por isso o pensamento computacional pode ser encarado como uma competência intelectual básica, como ler, escrever e conhecer as operações aritméticas, fazendo parte da área cognitiva do indivíduo. Sobre a abstração, usada pelo PC, é uma maneira de ampliar esse intelecto, sabendo lidar com a complexidade de um problema (COUNCIL, 2010), procurando a tecnologia mais adequada, sendo necessário aprender, encontrar alternativas e avaliar qual a melhor estratégia para validar sua teoria.

O pensamento computacional não é algo para os computadores, não é uma tentativa de trazer o pensamento para a máquina, mas sim para quem a comanda. Esse é o papel do cientista da computação, pesquisar, encontrar ou criar formas eficientes de usar a tecnologia para solucionar problemas, improvisando no que for possível, e não para "robotizar" o pensamento humano (WING, 2006). Por isso Araújo, Silveira e Mattos (2018) já citam que o PC deve ser prioridade para combater o atraso tecnológico presenciado no Brasil e a própria SBC começou a incluir essa temática dentro dos cursos de graduação, visto em (ZORZO et al., 2017).

O professor não pode se prender à ideia de que essa é uma comunicação de mão única, onde ele apenas passa o conhecimento, ele também pode aprender com seus alunos, aprimorando sua didática, identificando onde a computação pode ser usada dentro do contexto da sala de aula e em sociedade. O educador também deve incentivar o desejo pelo conhecimento nos alunos, estimular que eles aprendam mais incluindo e interagindo com todos na participação das aulas, utilizando abordagens que provoquem o envolvimento deles com o assunto de aula, tornando-os mais colaborativos. Usar projetos interdisciplinares é uma boa saída para isso, onde os alunos poderão utilizar os conceitos da ciência da computação em outras áreas e assim exercitar suas competências e visualizar novos problemas de formas diferentes (ISTE, 2018).

A curiosidade deve ser parte do aprendizado de um aluno envolvido com o pensamento computacional, ele tem que ser autônomo em seus objetivos, se comunicar com pessoas de diferentes costumes e pensamentos, só assim ele irá ampliar seus horizontes. Usar a tecnologia de forma consciente e justa, criando o seu papel de um bom cidadão digital e ajudar a construir conhecimento baseado no cotidiano, além de divulgar cada nova solução criada, aproveitando a tecnologia que interliga pessoas de diversas partes do mundo para isso. Sempre buscar a inovação, seja na forma de desenvolver algo, na hora de modelar, criar uma solução ou até mesmo propor um problema, testar e validar algo já existente (ISTE, 2016).

O uso dessa abordagem em salas de aula e no cotidiano dos cursos de computação já mostra resultados (SCAICO et al., 2012); (VAHLICK et al., 2016) demonstrando que o pensamento computacional estimula o aprendizado dos alunos, o que pode amenizar as dificuldades enfrentadas por alunos do ensino médio que estão chegando na graduação com um déficit de habilidades básicas, como visto em (BARCELOS; TAROUÇO; BERCHT, 2009); (MALTEMPI; VALENTE, 2000); (RIBAS; BIANCO; LAHM, 2016).



### 3 Dificuldades no Aprendizado em Programação no Ensino Superior

Em sua tese de doutorado, Medeiros (2019) realizou uma extensa Revisão Sistemática da Literatura (RSL) sobre as dificuldades de estudantes iniciantes em programação no ensino superior brasileiro. A RSL foi composta por uma pesquisa manual em periódicos da Revista Novas Tecnologias na Educação (RENOTE), na Revista Brasileira de Informática na Educação (RBIE), Workshop de Educação em Computação (WEI), Workshop de Informática na Educação (WIE), Workshop de Ensino em Pensamento Computacional, Algoritmos e Programação (WAlgProg) e no Simpósio Brasileiro de Informática na Educação (SBIE), complementada por uma busca automática usando uma *string* de busca usada no Google Scholar, definida como: (“aprendizagem de programação” OR “ensino de programação”) AND (“iniciantes” OR “introdução”).

O autor procurou responder, com sua pesquisa, as seguintes questões: quais habilidades e conhecimentos prévios são fundamentais para um aluno iniciante aprender a programar?; que desafios os estudantes iniciantes encontram para aprender a programar?; e quais desafios os professores encontram ao ensinar a programação introdutória?

Medeiros (2019) também utilizou alguns critérios em sua pesquisa, como: artigos que não abordam as questões da pesquisa, muito curtos, publicados em eventos sem relevância nacional, escritos pelos mesmos pesquisadores, e pesquisas que não foram realizadas no Brasil. Após aplicar a filtragem dos artigos, Medeiros (2019) realizou o seu estudo com os 84 artigos que atenderam todos os critérios da sua pesquisa.

Medeiros (2019) categorizou artigos brasileiros sobre o aprendizado de programação no ensino superior baseados em habilidades prévias necessárias para o seu aprendizado, essas habilidades foram divididas em duas áreas: uma sobre os conhecimentos e competências básicas da própria programação e outra referente às competências da educação básica e cognitiva do indivíduo.

Nas habilidades referentes à programação vemos que a principal habilidade necessária para a aprendizagem de programação, de acordo com a literatura brasileira é o raciocínio lógico, citado em 24% das publicações (MEDEIROS, 2019). Vemos que a abstração, noções matemáticas e a resolução de problemas, com 15%, 14% e 13% respectivamente, são apoiadas pela necessidade do raciocínio lógico. Apenas 7% das publicações consideram que ter algum conhecimento prévio sobre programação seja primordial para o aprendizado, em relação às outras habilidades (MEDEIROS, 2019).

Em relação às habilidades educacionais gerais, (MEDEIROS, 2019) identificou as seguintes categorias: conhecimento básico em português e interpretação de texto, criatividade, pensamento crítico, curiosidade e habilidade de discussão, hábito de estudo, concentração e memorização, e maturidade. Cada uma dessas categorias são encontradas em 7%, 3%, 2%, 2%, 2% e 1%, respectivamente, dos artigos usados por Medeiros (2019) na sua RSL.

### 3.1 Problemas identificados pelos professores

Medeiros (2019) também mapeou os artigos que citam as dificuldades encontradas pelos professores que lecionam programação. Em sua pesquisa, ele identificou os seguintes desafios: métodos e ferramentas, escalabilidade e infraestrutura, estimular a motivação dos alunos, comunicação com os alunos, linguagem de programação, currículo e sequências instrucionais, déficit no conhecimento matemático dos alunos e a valorização do erro. 56% dos artigos brasileiros apontam que os professores têm dificuldades em encontrar métodos e ferramentas comprovadamente eficazes (MEDEIROS, 2019).

Problemas com a infraestrutura das IES são comentados em 27% das publicações brasileiras sobre o tema, pela grande quantidade de alunos por turma e baixa quantidade de laboratórios de informática e monitores para os auxiliarem (MEDEIROS, 2019). Nas publicações brasileiras, foi identificado que o erro durante o aprendizado é pouco visto como forma do processo, usado em 6% dos trabalhos (MEDEIROS, 2019). Os professores também encontram resistências e entraves para ajustar as ementas das disciplinas introdutórias de programação para atender as recomendações da SBC (MEDEIROS, 2019).

### 3.2 Problemas identificados pelos alunos

Quando avaliadas as dificuldades dos alunos com o aprendizado de programação, Medeiros (2019) categorizou-as em: formalização do problema, expressão da solução, execução da solução e avaliação, conhecimento de base e comportamento. 12% das publicações citam que os alunos possuem dificuldades em pensar algorítmicamente e raciocinar de forma lógica, e 21% sobre como abstrair esses problemas. Analisando esses dados as maiores dificuldades estão em pontos chave do PC.

Sobre a formalização do problema, Medeiros (2019) aponta a resolução de problema, abstração, pensamento algorítmico e raciocínio lógico, como as principais dificuldades dos alunos no aprendizado de programação, com aproximadamente 25%, 21% e 12% respectivamente.

Sobre as expressões da solução, Medeiros (2019) identificou que em 15% dos casos a sintaxe da linguagem de programação lecionada pode ser impactante para os alunos, podendo ser um indício dos problemas com a depuração, dentro da execução da solução e avaliação, vista em 15% das publicações. Já nos conhecimentos de base, interpretação de texto e noções matemáticas, com 15% e 10% respectivamente, são obstáculos para os alunos, enquanto se manterem motivados com o aprendizado é um comportamento abordado em 30% dos trabalhos brasileiros (MEDEIROS, 2019).

### 3.3 Oportunidades

Como visto nas seções anteriores, algo precisa ser feito para que a dificuldade dos alunos e professores com o aprendizado de programação seja mitigada. Quanto aos problemas encontrados pelos alunos, no aprendizado de programação, a resolução de problemas, sintaxe da linguagem de programação, depuração, interpretação de texto e motivação são as principais dificuldades dos alunos (MEDEIROS, 2019).

Esses pontos são abordados e ditos como pontos chave para o pensamento computacional, defendidos como competências despertadas por ele e que devem ser mantidas por toda vida como parte do processo cognitivo, ampliando o intelecto do indivíduo, dando abertura para que possamos utilizá-lo em sala de aula (WING, 2006); (ISTE, 2018); (ISTE, 2016).

Assim, a proposta deste trabalho é verificar se a aplicação do pensamento computacional no ensino superior traz benefícios reais no aprendizado e na afetividade dos alunos com a programação, reduzindo suas dificuldades e os ajudando a desenvolver as competências necessárias ao aprendizado de programação.

## 4 Trabalhos Relacionados

Há poucos trabalhos na literatura que investigam PC no ensino superior no Brasil, em comparação ao ensino fundamental e médio, como mostram as RSL realizadas por Zanetti, Borges e Ricarte (2016); Medeiros (2019). Estas RSL trazem um levantamento das práticas educacionais envolvendo o pensamento computacional nos ensinos fundamental, médio/técnico e superior, identificando que o grande esforço dos pesquisadores e educadores que estão utilizando essa habilidade está no ensino médio.

Além de ter os trabalhos de Zanetti, Borges e Ricarte (2016) e Medeiros (2019) como referência, foi feito um levantamento dos artigos brasileiros que abordam os temas de pensamento computacional na educação básica, para estudar e conhecer as aplicações e formas com que o PC está sendo tratado em salas de aula; o pensamento computacional no ensino superior, como ele é tratado e aplicado dentro dos cursos superiores em computação, se focam apenas nos alunos, professores ou ambos; e sobre as dificuldades no aprendizado de programação no ensino superior, para focar nos problemas identificados por alunos e professores nessa troca de conhecimento. Para isso foi feito um levantamento dos artigos mais recentes, desde a publicação de Wing (2006), sobre o tema em anais de congressos e simpósios referentes à Educação em Computação, como o Workshop de Informática na Escola (WIE) e Congresso Brasileiro de Informática na Educação (CBIE).

Bosse, Gerosa (2015); Costa e Bublitz (2013); Fassbinder, Paula e Araújo (2012); Hoed (2016); Piva Jr. e Freitas (2010); MEC (1996); Ortiz, Oliveira e Pereira (2018); Peixoto *et al.* (2013); Pontual Falcão *et al.* (2018); Scaico *et al.* (2012); Vahldick *et al.* (2016); Wing (2016); Csizmadia *et al.* (2015), são estudos envolvendo alunos de cursos de computação no ensino superior. Dentre estes, Wing (2016); Csizmadia *et al.* (2015); ISTE (2011); ISTE (2018); Council (2010); ISTE (2016); Vahldick *et al.* (2016); Scaico *et al.* (2012); Ortiz, Oliveira e Pereira (2018); Pontual Falcão *et al.* (2018); Zorzo *et al.* (2017) abordam especificamente o pensamento computacional no ensino superior.

De forma mais geral, Costa e Bublitz (2013); Hoed (2016); MEC (1996); Scaico *et al.* (2012) mostram dados e estudos sobre a evasão e a retenção dos alunos dos cursos superiores de computação no Brasil, expondo as taxas e perfis daqueles que evadem de seus cursos de origem. Costa e Bublitz (2013) mostram que o desempenho nas disciplinas de programação pode ser um reflexo da infraestrutura da universidade e outros motivos alheios ao ambiente de sala de aula. Piva Jr. e Freitas (2010); Zanini e Raabe (2012) se mostram preocupados com a pouca habilidade em interpretação

de texto apresentada pelos alunos da graduação.

Mais especificamente falando de Pensamento Computacional, Vahldick *et al.* (2016) trazem uma experiência na criação e uso de um jogo focado no desenvolvimento das habilidades do PC com alunos do ensino superior de uma universidade portuguesa, apoiando o ensino da ementa das disciplinas de programação, mostrando que o uso dessa abordagem refletiu no interesse, na dedicação e no aprendizado dos alunos. Barcelos, Tarouco e Bercht (2009); Maltempi e Valente (2000); Ribas, Bianco e Lahm (2016) argumentam que o nível de conhecimento em raciocínio lógico dos ingressantes na graduação está cada vez menor, provavelmente devido a uma falta de estímulo e desenvolvimento na educação básica, podendo ser uma das causas para o insucesso durante as disciplinas de programação. Scaico *et al.* (2012) apresentam uma aplicação de tutores durante o ensino de programação com alunos do curso de Licenciatura em Computação, usando um grupo de controle para fazer uma comparação dos resultados, encontrando um impacto positivo considerável tanto no número de aprovados como na média das notas da turma.

De particular relevância para o presente trabalho, Pontual Falcão *et al.* (2018) propõem a inserção da temática do PC dentro do curso de LC da UFRPE para os futuros alunos de programação. Durante a criação do novo Projeto Pedagógico do Curso, foi criada e incluída em sua matriz curricular a disciplina de Pensamento Computacional, antes dos alunos terem contato com a programação, sendo obrigatória já no primeiro semestre do curso. Com o primeiro ano de experiência da disciplina de pensamento computacional no curso de LC da UFRPE, no segundo semestre de 2018 e primeiro semestre de 2019, é necessária uma avaliação comparativa dos novos alunos e os que cursaram programação sem esse contato prévio com PC na universidade, proposta pela presente pesquisa.

Além disso, poucas pesquisas sobre pensamento computacional consideram os dados socioculturais dos alunos para identificarem possíveis padrões que causem ou pré-disponham a evasão dos cursos de Computação Ortiz, Oliveira e Pereira (2018), sendo um ponto carente de atenção para entender melhor perspectivas dos alunos com o curso.

Araujo *et al.* (2013); Sirotheau *et al.* (2012); Guedes (2014); Gomes *et al.* (2015) focam mais no estudo que o ensino conteúdo das aulas, a didática dos professor e as instalações da IES, enquanto Santos *et al.* (2015); Iepsen, Bercht e Reategui (2012); Lima e Leal (2013); França e Amaral (2013); Ambrósio (2011); Giraffa e Mora (2013), levam mais em consideração a perspectiva do aluno quanto o aprendizado nas disciplinas de programação e algoritmos, mas todos investigam as dificuldades no aprendizado de programação no ensino superior.

Este trabalho se destaca dos demais devido ao estudo das vantagens do con-

tato no pensamento computacional no ensino superior, investigando o seu impacto nas questões afetivas que os alunos desenvolvem com a disciplina de programação. Na Tabela 1 é possível encontrar um resumo dessas informações.

A tabela abaixo usa alguns marcadores para suas colunas, eles representam tipos de relacionamento que os mesmos possuem com este trabalho.

- TR1: Pensamento Computacional na Educação Básica;
- TR2: Pensamento Computacional no Ensino Superior;
- TR3: Evasão nos cursos de computação;
- TR4: Dificuldades no Aprendizado de Programação no Ensino Superior.

Tabela 1 – Resumo dos trabalhos relacionados

<b>Artigos</b>	<b>TR1</b>	<b>TR2</b>	<b>TR3</b>	<b>TR4</b>
Este trabalho	-	X	-	-
(AMBRÓSIO, 2011)	-	-	-	X
(ARAUJO et al., 2013)	-	-	-	X
(BARCELOS; TAROUCO; BERCHT, 2009)	-	-	-	-
(BOSSE; GEROSA, 2015)	-	-	-	-
(COSTA; BUBLITZ, 2013)	-	-	X	-
(COUNCIL, 2010)	X	X	-	-
(CSIZMADIA et al., 2015)	X	X	-	-
(FASSBINDER; PAULA; ARAÚJO, 2012)	-	-	-	-
(FRANÇA; AMARAL, 2013)	-	-	-	X
(GIRAFFA; MORA, 2013)	-	-	-	X
(GOMES et al., 2015)	-	-	-	X
(GUEDES, 2014)	-	-	-	X
(HOED, 2016)	-	-	X	-
(IEPSEN; BERCHT; REATEGUI, 2012)	-	-	-	X
(ISTE, 2011)	X	X	-	-
(ISTE, 2018)	X	X	-	-
(ISTE, 2016)	X	X	-	-
(LIMA; LEAL, 2013)	-	-	-	X
(MEC, 1996)	-	-	X	-
(MEDEIROS, 2019)	X	X	-	X
(PAPERT, 1980)	X	X	-	-
(ORTIZ; OLIVEIRA; PEREIRA, 2018)	-	X	-	-
(PEIXOTO et al., 2013)	-	-	-	-
(PIVA JR; FREITAS, 2010)	-	-	-	-
(PONTUAL FALCÃO et al., 2018)	-	X	-	-
(SANTOS et al., 2015)	-	-	-	X
(SCAICO et al., 2012)	-	X	X	-
(SIROTHEAU et al., 2012)	-	-	-	X
(VAHLDICK et al., 2016)	-	X	-	-
(WING, 2006)	-	X	-	-
(ZANINI; RAABE, 2012)	-	-	-	-
(ZORZO et al., 2017)	-	X	-	-

Fonte: este trabalho

## 5 Metodologia

Essa pesquisa foi realizada tendo como parâmetro o contexto do curso de Licenciatura em Computação da Universidade Federal Rural de Pernambuco (UFRPE), que introduziu na sua matriz curricular no segundo semestre do ano letivo de 2018 a disciplina de Pensamento Computacional como uma disciplina do primeiro semestre para os alunos ingressantes no curso, como forma de ajudar os alunos a se prepararem, e sendo pré-requisito, para cursarem programação. Participaram deste estudo os estudantes que cursaram a disciplina de Pensamento Computacional, os professores que lecionaram programação desde a mudança na matriz curricular e, pelo questionário ser feito *online* ele foi distribuído para alunos de outras IES brasileiras, porém só obtivemos respostas dos institutos Instituto Federal do Rio Grande do Norte (IFRN) e Instituto Federal do Triângulo Mineiro (IFTM), e as universidades Universidade Federal do Paraná (UFPR), Universidade Federal da Bahia (UFBA) e Universidade Federal do Sul da Bahia (UFSB), sendo respeitados os perfis de alunos identificados dentro da própria UFRPE, para investigar o impacto do contato prévio com o PC no aprendizado de programação desses alunos. Essas IES, além da UFRPE, foram escolhidas por também abordarem o Pensamento Computacional dentro dos seus respectivos cursos de Licenciatura em Computação.

Ao todo foram contatados, via *e-mail*, vários representantes de cursos de Licenciatura em Computação pelo país, como dos institutos Instituto Federal de Brasília (IFB), Instituto Federal da Bahia (IFBA), Instituto Federal Farroupilha (IFFar), Instituto Federal de Minas Gerais (IFMG), IFRN, IFTM, Instituto Federal do Tocantis (IFTO), e das universidades UFBA, Universidade Federal Rural do Semi-Árido (UFERSA), Universidade Federal de Juiz de Fora (UFJF), Universidade Federal da Paraíba (UFPB), UFPR, Universidade Federal Rural da Amazônia (UFRA), Universidade Federal do Sul da Bahia (UFSB) e Universidade Tecnológica Federal do Paraná (UTFPR).

Eles foram informados sobre a pesquisa e o seu anonimato foi garantido, não foi coletado nenhum dado que pudesse identificá-los, e dando consentimento para que os dados por eles fornecidos fossem utilizados neste trabalho.

Maiores detalhes de como foi conduzida essa pesquisa são descritos a seguir.

### 1. Revisão bibliográfica

Foi feito um estudo do estado da arte sobre o tema principal, investigando o impacto que o pensamento computacional pode proporcionar no aprendizado de programação e os temas que o apoiam, como os possíveis reflexos que essa experiência pode trazer nas questões afetivas dos alunos com a disciplina ou



com o curso, e se reduziria as chances deles evadirem do curso, visando uma relevância para a ciência e pesquisa brasileira.

As principais fontes usadas neste trabalho são (WING, 2006), por trazer o tema do Pensamento Computacional como área de pesquisa para a ciência da computação, (MEDEIROS, 2019); (ZANETTI; BORGES; RICARTE, 2016) por suas RSL, fazendo um levantamento das práticas educacionais sobre o pensamento computacional no ensino superior. (PONTUAL FALCÃO et al., 2018) por trazerem o contexto do PC para o curso de LC da UFRPE. (MEC, 1996); (HOED, 2016) por definirem e trazerem maiores informações sobre a evasão dos alunos no ensino superior, e (ORTIZ; OLIVEIRA; PEREIRA, 2018) por revelarem que os dados socioculturais dos alunos não são levados em consideração em muitas pesquisas.

## 2. Realizar entrevistas e colher dados com professores

Os professores de programação foram entrevistados de uma forma semiestruturada para conhecer as suas percepções sobre o desempenho dos alunos com a disciplina de programação.

## 3. Consolidar dados colhidos com professores

Foi realizada uma análise com os dados colhidos com os professores. Esses dados foram colhidos após uma entrevista presencial feita com os professores, onde eles exprimiram suas opiniões sobre o pensamento computacional e a experiência deles com os alunos que tiveram, ou não, esse contato, sobre suas expectativas em relação a essa abordagem e suas opiniões pessoais. Tudo foi coletado em forma de áudio gravado, com o consentimento dos entrevistados, e analisado os pontos em que as opiniões dos professores convergem ou divergem.

## 4. Planejar pesquisa com os dois perfis de alunos

Foram identificados os dois perfis de alunos abaixo e elaborado um questionário para cada, com suas particularidades, e aplicados para cada um dos perfis. Os questionários criados estão disponíveis nos Apêndices A e B.

- Alunos perfil 1 (AP1): alunos que tiveram contato com o PC em algum momento antes de programação, e;
- Alunos perfil 2 (AP2): alunos que tiveram contato com o PC depois de cursarem programação.

## 5. Aplicar os questionários aos alunos

Foram realizados os questionários com os alunos dos perfis AP1 e AP2. Esses questionários foram distribuídos online, utilizando o Google Forms, e os alunos

foram convidados a participar através dos grupos nas redes sociais e listas de *e-mails*, dos cursos de LC da UFRPE e de outras IES supracitadas, respeitando os perfis AP1 e AP2, onde os alunos trocam informações e conhecimento entre si.

#### **6. Consolidar dados colhidos com alunos**

Os dados colhidos com a aplicação dos questionários dos AP1 e AP2 foram analisados e identificados os pontos mais importantes fornecidos pelos alunos.

## 6 Resultados

Nesta seção serão apresentados os resultados deste trabalho. Esses resultados são oriundos das entrevistas feitas com os professores identificados durante a formulação da metodologia deste trabalho, e dos questionários feitos com os alunos do curso de Licenciatura de Computação.

### 6.1 Entrevistas com alunos do perfil AP1

Nesta seção são mostrados os resultados obtidos após a aplicação do questionário para os alunos do perfil 1 (AP1) (Apêndice A), identificado para esta pesquisa.

- Alunos perfil 1 (AP1): alunos ingressados no curso de LC e que tiveram contato com PC antes de cursarem Programação;

Tabela 2 – Distribuição dos participantes AP1 por IES

<b>IES</b>	<b>Número de participantes</b>
UFRPE	9
IFRN	6
UFPR	2
IFTM	2
UFBA	1

Fonte: este trabalho

Esse questionário foi feito de forma *online* e, por isso, foi respondido por pessoas de várias IES, sendo a UFRPE como o foco principal deste trabalho, e foram obtidas um total de 20 respostas, distribuídas conforme a tabela 2.

Tabela 3 – Motivos para alunos AP1 cursarem Licenciatura em Computação

<b>Motivos para cursar Licenciatura em Computação</b>	<b>% de respostas</b>
Aptidão e gosto pela computação	31,6%
Desejo pela licenciatura	21,1%
Cursar uma graduação em um horário que se adequa a minha realidade	10,5%
Não consegui ingressar em outros cursos da área	5,3%
Desejo pela licenciatura, Aptidão e gosto pela computação	10,5%
Aptidão e gosto pela computação, Cursar uma graduação em um horário que se adequa a minha realidade	5,3%
Desejo pela licenciatura, Aptidão e gosto pela computação, Cursar uma graduação em um horário que se adequa a minha realidade	10,5%
Curso é ofertado na própria cidade minimizando custos de transporte	5,3%

Fonte: este trabalho

Para a questão 2, tabela 3, sobre os motivos de cursar a Licenciatura em Computação, como respostas únicas, 31,6% dos participantes responderam que têm aptidão pela computação, desejo pela licenciatura com 21,1%, cursar uma graduação em um horário que se adequa necessidades, 10,5%, e que não conseguiram ingressar em outros cursos da área, 5,3%. Como respostas mistas, combinando uma ou mais respostas individuais, o desejo pela licenciatura e a aptidão pela computação, com 10,5%, aptidão pela computação e adequar aos horários com 5,3%, e desejo pela licenciatura, aptidão pela computação e adequar aos horários, foram 10,5%. Também atenta-se para uma única resposta sobre que o curso de Licenciatura em Computação é ofertado na própria cidade do participante, minimizando custos de transporte, que inviabilizaria o desejo de cursar computação. Vemos que o desejo pela licenciatura e a aptidão pela computação são os principais motivos para que esses alunos ingressem no curso de Licenciatura em Computação.

Na questão 3, sobre se os participantes mantêm dupla jornada, ou seja, trabalham e estudam em horários diferentes, 78,9% responderam que sim e 21,1% responderam que não, o que pode representar que possa haver um desgaste ao longo do dia e o rendimento em sala de aula, a depender dos horários. Esse sentimento é confirmado pela questão 3.1, sobre a dupla jornada interferir na dedicação e aprendizado de programação, onde 75% responderam que sim e 25% que não.

Com a questão 4, sobre a possível evasão do curso ou se já evadiu, tendo que ser reintegrado à graduação, 31,6% responderam que sim, e 68,4% que não. Os que

foram reintegrados, ou se já pensaram em evadir, na questão 4.1, 75% responderam que o motivo da evasão foi conciliar a vida profissional com os estudos, enquanto 25% responderam que estão na sua primeira graduação.

Quando questionados sobre algum receio anterior a entrar na graduação por terem que aprender programação, na questão 5, 47,4% responderam que sim e 52,6% que não. Dentre os que afirmaram ter algum receio, na questão 5.1, 76,9% se sentiram mais tranquilizados ao saberem que teriam uma disciplina introdutória antes de fato cursarem programação, enquanto 23,1% disseram que não.

94,7% dos participantes acreditam que o Pensamento Computacional aumentou sua confiança e preparação para aprender programação, enquanto 5,3% responderam que não, na questão 6. Todos os participantes responderam na questão 7 que acreditam que o Pensamento Computacional contribui com os desafios interdisciplinares do cotidiano.

Tabela 4 – Níveis de raciocínio lógico dos alunos AP1

<b>Nível de raciocínio lógico</b>	<b>% de respostas</b>
Nenhum	0%
Ruim	31,6%
Regular	36,8%
Bom	26,3%
Muito bom	5,3%

Fonte: este trabalho

Quando questionados sobre o nível do raciocínio lógico, na questão 8, tabela 4, antes de entrarem na graduação, 36,8% consideram como regular, 31,6% como ruim, 26,3% como bom e 5,3% como muito bom.

Tabela 5 – Níveis de pensamento algorítmico dos alunos AP1

<b>Nível de pensamento algorítmico</b>	<b>% de respostas</b>
Nenhum	15,8%
Ruim	26,3%
Regular	42,1%
Bom	15,8%
Muito bom	0%

Fonte: este trabalho

Na questão 9, tabela 5, sobre o conhecimento relacionado ao pensamento algorítmico antes da graduação, 42,1% consideram como regular, 26,3% como ruim, e nenhum ou bom tiveram ambas 15,8% das respostas.

Ao serem questionados se cursar Pensamento Computacional os auxiliaram a desenvolver ou melhorar os seus níveis de raciocínio lógico e algorítmico, na questão

10, 89,5% responderam que sim e 10,5% responderam que não.

Foi deixado um espaço em aberto para que os participantes incluíssem suas próprias opiniões sobre o Pensamento Computacional e três deles deram opiniões bem interessantes sobre o uso e a importância do PC dentro da graduação em computação. Cada resposta aqui listada foi dada diretamente por um aluno que respondeu este questionário (Q1) para alunos do AP1, identificados como P (participante) com uma numeração, garantindo seu anonimato.

- *”Com pensamento computacional ficou fácil entender lógica de programação, e com isso, ficou tudo mais fácil aprender programação”, Q1P04;*
- *”Fundamental para quem entra em uma graduação na área de computação, pois direciona o discente de uma forma mais gradual”, Q1P06, e;*
- *”Sobre não pensar no problema como um todo, mas dividir em processos e ir avançando aos poucos para solucionar”, Q1P15.*

## 6.2 Entrevistas com alunos do perfil AP2

Nesta seção são mostrados os resultados obtidos após a aplicação do questionário para os alunos do perfil 2 (AP2) (Apêndice B), identificado para esta pesquisa.

- Alunos perfil 2 (AP2): alunos ingressados no curso de LC e que só tiveram contato com PC após cursarem Programação.

Tabela 6 – Distribuição dos participantes AP2 por IES

<b>IES</b>	<b>Número de participantes</b>
UFRPE	9
IFRN	5
UFPR	5
UFBA	3
IFTM	3
UFSB	1

Fonte: este trabalho

Esse questionário foi feito de forma *online* e, por isso, foi respondido por pessoas de várias IES, sendo a UFRPE como o foco principal deste trabalho, e foi obtido um total de 26 respostas, distribuídas conforme a tabela 6.

Tabela 7 – Motivos para alunos AP2 cursar Licenciatura em Computação

<b>Motivos para cursar Licenciatura em Computação</b>	<b>% de respostas</b>
Desejo pela licenciatura	8%
Aptidão e gosto pela computação	4%
Cursar uma graduação em um horário que se adequa a minha realidade	12%
Não consegui ingressar em outros cursos da área	12%
Desejo pela licenciatura, Aptidão e gosto pela computação	20%
Aptidão e gosto pela computação, Cursar uma graduação em um horário que se adequa a minha realidade	12%
Cursar uma graduação em um horário que se adequa a minha realidade, Não consegui ingressar em outros cursos da área	8%
Desejo pela licenciatura, Cursar uma graduação em um horário que se adequa a minha realidade, Aumentar a empregabilidade	4%
Aptidão e gosto pela computação, Não consegui ingressar em outros cursos da área, Cursar uma graduação em um horário que se adequa a minha realidade	4%
Cursar uma graduação em um horário que se adequa a minha realidade, Procurar nova carreira	4%

Fonte: este trabalho

Na questão 2, tabela 7, ao serem questionados pelos motivos que os fizeram cursar Licenciatura em Computação, os participantes se dividiram em dois tipos de respostas, únicas e compostas, sendo cursar uma graduação que se adequa aos horários e por não conseguirem ingressar em outros cursos da área, com 12% ambas, desejo pela licenciatura com 8%, e aptidão pela computação e outros com 4%, sendo que esse outros se justificou pelas disciplinas dos outros cursos não o agradarem. Já nas respostas compostas tivemos desejo pela licenciatura e aptidão pela computação com 20%, aptidão pela computação e buscando um horário mais adequado com 12%, um horário mais adequado e por não conseguirem ingressar em outros cursos da área com 8%, e com 4% para desejo pela licenciatura, horário e aumentar a empregabilidade, aptidão pela computação, não conseguirem ingressar em outros cursos e horário, horário e procurando uma nova carreira, respectivamente. Mostrando, assim como no questionário AP1, que o desejo pela licenciatura e a aptidão da computação como os principais motivos para esses alunos ingressarem no curso da Licenciatura em Computação.

Quando questionados se mantêm uma dupla jornada de estudos e trabalho, em horários distintos, na questão 3, 68% responderam que sim e 32% responderam que não. Dentre os que mantêm dupla jornada, 66,7% responderam que isso pode afetar a sua dedicação ao aprendizado de programação, enquanto 33,3% disseram que não.

Tabela 8 – Motivos para evasão dos alunos AP2

<b>Motivos para evasão</b>	<b>% de respostas</b>
Problemas em conciliar os estudos com a vida profissional	0%
Tive sérias dificuldades com programação	16,7%
Dificuldades em outras disciplinas durante o curso que me deixaram desmotivado(a)	0%
Problemas de saúde comigo ou algum parente próximo	0%
Vim de outra área e tive dificuldades de me adaptar à área de computação	0%
Problemas em conciliar os estudos com a vida profissional, Dificuldades em outras disciplinas durante o curso que me deixaram desmotivado(a), Problemas de saúde comigo ou algum parente próximo, Dificuldades em atender aos horários das aulas	16,7%
Problemas em conciliar os estudos com a vida profissional, Tive sérias dificuldades com programação, Vim de outra área e tive dificuldades de me adaptar à área de computação	16,7%
Problemas de saúde comigo ou algum parente próximo, Problemas em conciliar os estudos com a vida profissional, Vim de outra área e tive dificuldades de me adaptar à área de computação	16,7%
Tive sérias dificuldades com programação, Dificuldades em outras disciplinas durante o curso que me deixaram desmotivado(a)	16,7%
Nenhum	16,7%

Fonte: este trabalho

48% dos participantes responderam que já consideraram evadir ou já evadiram da graduação, precisando serem reintegrados, enquanto 52% disseram que não. Dentre os que responderam sim a questão 4, tabela 8, 16,7% responderam que tiveram problemas com programação. Problemas em conciliar estudos e vida profissional, dificuldades em outras disciplinas, problemas de saúde e problemas em atender aos horários das aulas, com 16,7%. Já os que tiveram problemas em conciliar os estudos com a vida profissional, dificuldades com programação e por virem de outras áreas, tiveram 16,7% das respostas. Problemas de saúde, conciliar estudos com a vida pro-



fissional e por virem de outras áreas, com 16,7%. Com problemas em programação e desmotivação por dificuldades em outras disciplinas, e por alegarem que nenhum problema sugerido estava relacionado a sua evasão, com 16,7% respectivamente.

Ao serem questionados se possuíam receio de cursar programação antes de entrarem na graduação, 56% disseram que não enquanto 44% disseram que sim. Os que disseram que tinham algum receio, 83,3% disseram que se desejarium cursar alguma disciplina introdutória à programação, enquanto 16,7% responderam que não.

Sabendo que os alunos do AP2 tiveram contato com o Pensamento Computacional após cursarem programação, eles foram questionados se esse contato com PC antes de programação os auxiliaria, deixando-os melhores preparados e confiantes: 96% disseram que sim e 4% que não. Este resultado é um ponto crucial para este trabalho, dado que os próprios alunos consideram que abordar o Pensamento Computacional antes de aprenderem a programar e desenvolver *softwares* é algo válido.

Seguindo com o intuito do Pensamento Computacional auxiliar as pessoas no mundo interdisciplinar cotidiano, 92% responderam que concordam com esse pensamento e 8% que não.

Tabela 9 – Níveis de raciocínio lógico dos alunos AP2

<b>Nível de raciocínio lógico</b>	<b>% de respostas</b>
Nenhum	4%
Ruim	16%
Regular	44%
Bom	20%
Muito bom	16%

Fonte: este trabalho

Sobre a autoavaliação do conhecimento sobre raciocínio lógico, tabela 9, 44% dos participantes disseram que consideram regular, 20% como bom, muito bom e ruim com 16% respectivamente, e 4% como nenhum.

Tabela 10 – Níveis de pensamento algorítmico dos alunos AP2

<b>Nível de pensamento algorítmico</b>	<b>% de respostas</b>
Nenhum	28%
Ruim	12%
Regular	44%
Bom	16%
Muito bom	0%

Fonte: este trabalho

E sobre o pensamento algorítmico, tabela 10, 44% como regular, 28% como nenhum, 16% como bom, e 12% como ruim.

Segundo 80% dos participantes, o Pensamento Computacional contribuiu para que desenvolvessem ou melhorassem o seu conhecimento em raciocínio lógico e pensamento algorítmico, sendo um *feedback* muito bom para que dar continuidade no uso do PC na graduação. Já 20% disseram que isso não aconteceu com eles.

Ao final do questionário os participantes poderiam expressar suas próprias opiniões sobre o uso do Pensamento Computacional na graduação. Cada resposta aqui listada foi dada diretamente por um aluno que respondeu este questionário (Q2) para alunos do AP2, identificados como P (participante) com uma numeração, garantindo seu anonimato.

- Houveram sugestões para a disciplina em si, para abordar mais elementos da programação como uma fase preparatória, como em *"... trabalhar com pseudocódigo poderia resolver este problema para quem nunca teve contato com programação, ao final da cadeira de Pensamento Computacional"*, Q2P09;
- Também um dos participantes comentou que o PC reduziu suas incertezas com assuntos que envolvessem matemática e raciocínio *"Eu sempre tive muito receio e muito "medo"de tudo que envolvesse matemática e raciocínio lógico... Comecei a enxergar os desafios lógicos como algo da rotina..."*, Q2P14";
- Q2P19 comenta que o pensamento computacional é uma habilidade e não simplesmente lógica de programação, sendo uma abordagem mais profunda e sobre sua dupla jornada de estudos e trabalho, pois já ingressou no curso sabendo programar, acredita que a disciplina de PC seria de grande ajuda para seus colegas de turma que não tinham visto programação ainda na época, e um terceiro participante, Q2P23, comentou que muitos conceitos que, segundo ele, poderiam ajudar na programação são vistos em disciplinas posteriores à programação, o que pode ser um fator as dificuldades encontradas;
- Um dos tópicos mais interessantes levantados pelo participante Q2P26, onde comentou que o pensamento computacional pode gerar no aluno a decisão de seguir na graduação e na área, já que o ele está diretamente ligado às áreas de atuação da ciência da computação e leva-os a conhecer um pouco de tudo, identificando se essa área o agrada ou não.

### 6.3 Entrevistas com professores

Para este trabalho foram identificados dois perfis de professores que interessam para esta pesquisa, que lecionou a disciplina de programação com a matriz curricular antiga, antes da inserção do Pensamento Computacional, e que lecionou após a

modificação da matriz curricular, com a inserção do Pensamento Computacional. Foram identificados dois professores com estes perfis juntamente com a coordenação do curso de Licenciatura em Computação da UFRPE e serão identificados como:

A : que lecionou programação na matriz curricular antiga;

B : que lecionou na matriz curricular nova.

Um ponto precisa ser exposto antes dos resultados serem mostrados, o professor A lecionou programação apenas na matriz curricular antiga, enquanto o professor B lecionou a disciplina nos dois momentos, tanto antes como após a mudança da matriz.

A entrevista semiestruturada feita com os dois professores, em momentos distintos, foi feita questionando-os sobre alguns pontos para esta pesquisa e suas respostas são encontradas abaixo.

- **Houve melhora dos alunos que cursaram programação depois de PC, em relação aos que cursaram antes de PC?**

Devido ao pouco tempo e avanço dentro do curso, visto que a primeira turma de PC do curso está hoje no terceiro semestre, ainda não é conclusivo se o PC trouxe melhoras visíveis, talvez quando eles avançarem mais será mais perceptível se o PC realmente deu a eles os benefícios esperados, sendo essa a visão do professor A quando ele diz que *"É bem difícil ter um sentimento de melhora garantido, a gente trabalha com casos, se a gente analisar uma turma e num outro semestre outra turma tiver outro comportamento, não tem como a gente generalizar"*, e B ao dizer *"Eu até perguntei pros próprios alunos o que eles acharam de PC, se contribui de fato com o aprendizado. A maioria até disse que contribui, mas quando vamos ver os resultados. Nas notas, eu não vi essa contribuição... Fica a dúvida se realmente contribuiu, eu acredito que é bom, que contribui"*.

O professor A também menciona questões como a retenção dos alunos em disciplinas de programação ser algo esperado, devido a dificuldade e a quebra do paradigma de um aluno vindo do ensino médio, e que o PC dá maiores chances aos alunos de terem uma base de conhecimentos mais consolidada para darem continuidade ao curso, *"Meu sentimento é de que com certeza ajuda, mas a gente sabe que existe uma retenção maior e dificuldade nesse tipo de conteúdo. Eu tenho relatos da primeira turma de PC que agora está em programação que eles estão tendo dificuldades. O meu ponto de vista é esse, é que esse sequencialmente, dá mais chances pro aluno ter um caminhar mais fluído, um aprendizado mais sólido... Cada vez mais a gente tem trabalhar não só a parte técnica, mas a motivação também."*

- **Essas dificuldades seriam com assuntos específicos de programação?**

Os alunos estão chegando na disciplina de Programação com dificuldades em compreender e assimilar os conteúdos que são passados, porém a perspectiva que o professor A tem é que isso melhore com num curto a médio prazo, *"Eu também sou supervisor de área e os professores me reportam esses acontecimentos... Eu estou otimista porque o perfil do aluno antigo é aquele que sai do ensino médio e já entra diretamente numa disciplina pesada como Programação"*, enquanto B acha que os níveis de proeficiência das turmas podem variar de semestre para semestre e isso pode influenciar num curto prazo de tempo para avaliar quantitativamente o quão beneficiador PC está sendo, *"Às vezes a gente pega uma turma que é muito boa e às vezes a gente pega uma turma que é fraca"*.

Há uma preocupação dos entrevistados para que sejam incluídos mais tópicos de programação na disciplina de PC, mesmo que mais perto do final do semestre, para que os alunos não cheguem tão imaturos à disciplina de Programação, usando linguagens mais amigáveis, como dito por A, *"Se o processo cognitivo do aluno de PC for indo de forma que ele faça a ponte com programação, quando ele sai do primeiro pro segundo período, ele vai ter um aproveitamento maior dos conceitos que ele viu. Se ele ficar muito limitado a visão de PC fora do escopo do ensino de programação, ele vai ter que gastar mais energia aprendendo novamente"*, e B, *"O feedback que eu tive dos alunos quando eles chegaram na minha disciplina de Programação 1 que eles não estavam programando, na disciplina de PC, que eles poderiam ter visto programação de fato. Nada como C ou Python, mas uma coisa como Portugal"*.

- **Essas dificuldades com programação estariam relacionadas à dificuldade com matemática?**

O fator dos alunos terem aptidões com a matemática e a programação tendem ser algo que caminham juntas, as habilidades com a matemática servem como sustentação às competências esperadas de bons alunos em programação, como dito por B, *"Chegam muitos alunos sem a base matemática e esses alunos não progridem. Os alunos que são bons em matemática, normalmente, vão ser bons em programação"*, e apoiado por A, *"Eu acho que tem semelhança, as duas ciências envolvem abstração... Vai ter uma hora que você vai ter que por o pensamento abstrato pra trabalhar, e no casamento de padrões. Têm semelhanças e têm diferenças também"*.

## 7 Conclusões e Trabalhos futuros

Nesta seção são expostas, discutidas e apresentadas as conclusões deste trabalho. Tendo como base de discussão os objetivos deste trabalho, aqui serão confrontadas as opiniões expressas pelos alunos, professores e com o que temos na literatura sobre o uso do Pensamento Computacional no ensino superior para alunos do curso de Licenciatura em Computação. Esta discussão se dá por tópicos, onde cada um remete aos objetivos do presente trabalho.

- **Examinar a percepção do aprendizado sob a visão dos estudantes e professores comparando o impacto do contato prévio com Pensamento Computacional**

Como dito previamente, este trabalho se baseia em dois perfis de alunos, aqueles que tiveram contato antes e aqueles que vivenciaram o Pensamento Computacional depois de cursarem Programação, AP1 e AP2 respectivamente.

Dentro do grupo de alunos do AP1, temos que 94,7% consideram que o Pensamento Computacional aumentou sua preparação para aprenderem programação, e 96% dos alunos do AP2 dizem que se sentiriam melhores preparados para cursarem programação se tivessem esse contato com o Pensamento Computacional previamente à Programação. No entanto, pela visão dos professores entrevistados, essa afirmação por parte dos estudantes não se reflete nas salas de aula de Programação, não sendo perceptíveis os ganhos no desempenho em notas desses alunos que vem do contato com o Pensamento Computacional.

Alguns pontos questionados aos alunos foram que se o Pensamento Computacional contribuiu para a melhora do raciocínio lógico e pensamento algorítmico dos estudantes, 89,5% dos alunos do AP1 e 80% dos alunos do AP2 afirmaram que o Pensamento Computacional os ajudou no raciocínio lógico e pensamento algorítmico. Esses pontos são de grande importância pois, (MEDEIROS, 2019), em sua tese, identificou em 12% dos artigos da literatura brasileira sobre o tema que esses pontos que dificultam o avanço dos alunos nas disciplinas e no aprendizado de Programação. Tomando o devido cuidado para não excluir a ideia de que errar não faz parte do processo de aprendizado, sendo visto em 6% das publicações brasileiras a respeito (MEDEIROS, 2019), sendo o déficit que os alunos têm ao chegar nas disciplinas de Programação o problema a ser tratado aqui.

Portanto, os professores entrevistados se mostram otimistas e que veem o Pensamento Computacional como uma ferramenta que auxiliem os alunos no aprendizado de Programação, porém ainda se mantém mais cautelosos em afirmarem

veemente se esse contato é benéfico e preferem aguardar o reflexo disso no desempenho dos alunos em sala de aula.

- **Identificar pontos de melhoria na disciplina de PC para favorecer o aprendizado de programação**

Com intuito de agregar um conteúdo que prepare melhor o aluno na disciplina de Pensamento Computacional, ambos, alunos e professores, fizeram sugestões em seus respectivos questionários e entrevistas, para que mais conteúdos de programação seja incluídos dentro da disciplina, de forma mais diluída ou um pouco mais abrangente.

Os professores defendem essa ideia ao argumentarem que se o aluno não for, de certo modo, cognitivamente preparado para programação desde o PC, ele terá um esforço maior para assimilar alguns conceitos em programação, e incentivando o uso de algumas linguagens mais lúdicas e isso, segundo um dos professores entrevistados, é algo que os próprios alunos dizem que sentiram falta durante a disciplina de PC.

Essa sugestão também foi encontrada nas respostas dos alunos AP2, onde comentaram que a disciplina de PC poderia envolver mais pseudocódigos, auxiliando aqueles que nunca tiveram contato com programação antes de ingressar ao curso e antecipando conceitos que são vistos em disciplina após programação, que sentiram que poderiam ter sido melhor aproveitados se fossem vistos antes da disciplina. Sendo uma forma de atacar o que é visto em 15% das publicações brasileiras sobre o impacto para os alunos ao se depararem com a sintaxe de alguma linguagem de programação (MEDEIROS, 2019).

- **Investigar as questões afetivas que os alunos desenvolvem com a disciplina de programação, comparando as visões dos que tiveram e dos que não tiveram contato com PC antes de cursar programação**

Ao questionar os alunos se eles tinham receio em aprender programação antes de ingressar na graduação, vemos que 47,4% e 44% dos alunos do AP1 e AP2, respectivamente, responderam que sim, e, logo, 94,7% dos alunos do AP1 e 96% dos alunos do AP2 se sentiram, ou sentiriam, mais confiantes com o contato com o Pensamento Computacional antes de cursarem programação, sendo um comportamento abordado em 30% dos trabalhos brasileiros (MEDEIROS, 2019).

Esse pensamento é reforçado pelas respostas dos alunos do AP2 quando, pelo menos, 50,1% responderam que um dos motivos para pensarem em evadir ou por terem evadido do curso ser dificuldades com programação. Ou seja, em torno da metade dos alunos que correm risco de evadirem do curso são por ques-

tões afetivas à disciplina de programação, confirmando o que é visto em (HOED, 2016).

Como visto nas respostas aos questionários pelos alunos dos dois tipos, AP1 e AP2, programação é uma disciplina que causa muito impacto ao chegar a um curso de computação, é um tema complexo e não é trivial o entendimento dele, por isso muitos chegam a considerar ou realmente abandonam o curso por causa de programação. Seja porque mantém uma dupla jornada, conciliando a vida profissional e os estudos, o que é uma realidade para 78,9% dos alunos AP1 e 68% dos alunos AP2, onde entre eles 75% e 66,7%, respectivamente, afirmam que essa necessidade de manter uma dupla jornada afeta diretamente sua dedicação e aprendizado a programação, mesmo entre aqueles que responderam que têm aptidão e gosto pela computação e que estudam um horário que se adequa melhor a sua realidade. E que a maioria dos alunos que responderam a pesquisa deste trabalho afirmam que ter uma disciplina ou um contato que os preparem mais para programação pode diminuir o impacto e a complexidade no aprendizado, acreditando que esse é o papel do pensamento computacional, já que, provavelmente, esses alunos não tiveram uma preparação para essa área durante o ensino médio, ponto esse que não foi contemplado neste trabalho.

## 7.1 Trabalhos futuros

Devido ao prazo de entrega deste trabalho, não foi possível fazer um acompanhamento mais prolongado dos alunos dos perfis AP1 e AP2, no curso de Licenciatura da Computação da UFRPE, e sua evolução no curso após o contato com o Pensamento Computacional, ficando então como um possível trabalho futuro acompanhar algumas turmas de novos ingressantes por mais semestres letivos e verificando o desempenho desses alunos de uma forma mais próxima, comparando notas e outras formas avaliativas, com a média histórica de turmas antes da inserção do Pensamento Computacional na matriz curricular.

Também devido ao prazo deste trabalho, não foi possível realizar uma entrevista mais aprofundada com outros professores além dos que foram entrevistados na UFRPE, o que pode ser um fator importante para aprofundar a pesquisa sobre o tema.

Levando em consideração os pontos apresentados neste trabalho, se faz benéfica a realização de um estudo para compreender a contextualização do Pensamento Computacional em outras IES brasileiras e avaliar o impacto no desempenho dos alunos nessas instituições.

Também foi sugerido pelos entrevistados por este trabalho a possibilidade de incluir tópicos do ensino de programação já na disciplina de Pensamento Computacio-

nal, como forma de preparar o aprendizado e o processo cognitivo desses alunos para programação, necessitando de um estudo para saber quais tópicos seriam esses e a melhor maneira de retratá-los dentro do contexto do Pensamento Computacional.



## Referências

- AMBRÓSIO, A. P. Programação de computadores: compreender as dificuldades de aprendizagem dos alunos. Universidade da Coruña, 2011.
- ARAUJO, E. C. de et al. O papel do hábito de estudo no desempenho do aluno de programação. 2013.
- BARCELOS, R.; TAROUÇO, L.; BERCHT, M. O uso de mobile learning no ensino de algoritmos. *RENOTE*, v. 7, n. 3, p. 327–337, 2009.
- BOSSE, Y.; GEROSA, M. A. As disciplinas de introdução à programação na usp: um estudo preliminar. v. 4, n. 1, p. 1389, 2015.
- COSTA, T.; BUBLITZ, F. Análise dos principais problemas que afetam alunos de programação: uma investigação empírica no estado da paraíba. 2013.
- COUNCIL, N. R. *Report of a Workshop on the Scope and Nature of Computational Thinking*. Washington, DC: The National Academies Press, 2010. ISBN 978-0-309-14957-0. Disponível em: <<https://www.nap.edu/catalog/12840/report-of-a-workshop-on-the-scope-and-nature-of-computational-thinking>>.
- CSIZMADIA, A. et al. *Computational thinking-A guide for teachers*. Computing At School, 2015.
- FASSBINDER, A. d. O.; PAULA, L. d.; ARAÚJO, J. C. D. Experiências no estímulo à prática de programação através do desenvolvimento de atividades extracurriculares relacionadas com as competências de conhecimentos. v. 32, 2012.
- FRANÇA, R. S. de; AMARAL, H. J. C. do. Mineração de dados na identificação de grupos de estudantes com dificuldades de aprendizagem no ensino de programação. *RENOTE*, v. 11, n. 1, 2013.
- GIRAFFA, M. M.; MORA, M. da costa. Evasão na disciplina de algoritmo e programação: um estudo a partir dos fatores intervenientes na perspectiva do aluno. In: *Congressos CLABES*. [S.l.: s.n.], 2013.
- GOMES, M. et al. Um estudo sobre erros em programação-reconhecendo as dificuldades de programadores iniciantes. In: *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*. [S.l.: s.n.], 2015. v. 4, n. 1, p. 1398.
- GUEDES, E. B. Um estudo observacional sobre a disciplina introdutória de programação. In: *Anais do Workshop de Informática na Escola*. [S.l.: s.n.], 2014. v. 20, n. 1, p. 552.
- HOED, R. M. Análise da evasão em cursos superiores: o caso da evasão em cursos superiores da área de computação. *Brasília, DF: Universidade de Brasília*, 2016.
- IEPSEN, E. F.; BERCHT, M.; REATEGUI, E. Detecção e tratamento do estado afetivo frustração do aluno na disciplina de algoritmos. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2012. v. 1, n. 1.

- ISTE. *Operational Definition of Computational Thinking for K–12 Education*. 2011.
- ISTE. *Standards for Students*. 2016.
- ISTE. *Standards for Educators: Computation Thinking Competencies*. 2018.
- LIMA, M. R. de; LEAL, M. C. Motivação discente no ensino-aprendizagem de programação de computadores. *Educação & Tecnologia*, v. 17, n. 1, 2013.
- MALTEMPI, M. V.; VALENTE, J. A. Melhorando e diversificando a aprendizagem via programação de computadores. 2000.
- MEC, D. retenção e evasão nos cursos de graduação em instituições de ensino superior públicas. *Avaliação: Revista de rede de avaliação institucional da educação superior*. Campinas, v. 1, n. 2, p. 55–65, 1996.
- MEDEIROS, R. P. Hello, world: uma análise sobre dificuldades no ensino e na aprendizagem de introdução à programação nas universidades. 2019.
- ORTIZ, J. dos S.; OLIVEIRA, C. M.; PEREIRA, R. Aspectos do contexto sociocultural dos alunos estão presentes nas pesquisas para ensinar pensamento computacional? v. 7, n. 1, p. 520, 2018.
- PAPERT, S. *Mindstorms: Children, Computers, and Powerful Ideas*. New York, NY, USA: Basic Books, Inc., 1980. ISBN 0-465-04627-4.
- PEIXOTO, M. et al. Uso de estratégias de aprendizagem e motivacionais pelos alunos em disciplinas de programação: Um estudo de caso na licenciatura em computação. 2013.
- PIVA JR, D.; FREITAS, R. L. Estratégias para melhorar os processos de abstração na disciplina de algoritmos. v. 1, n. 1, 2010.
- PONTUAL FALCÃO, T. et al. Currículo da licenciatura em computação: uma proposta alinhada às novas diretrizes e demandas contemporâneas. v. 7, n. 1, p. 1108, 2018.
- RIBAS, E.; BIANCO, G. D.; LAHM, R. A. Programação visual para introdução ao ensino de programação na educação superior: uma análise prática. *RENOTE*, v. 14, n. 2, 2016.
- SANTOS, A. et al. A importância do fator motivacional no processo ensino-aprendizagem de algoritmos e lógica de programação para alunos repetentes. In: *WEI-Workshop sobre Educação em Computação*. [S.l.: s.n.], 2015. p. 1–10.
- SCAICO, P. et al. Relato de um modelo de tutoria para programação baseado em experiências com ingressantes de um curso de licenciatura em computação. 2012.
- SIROTHEAU, S. et al. Aprendizagem de iniciantes em algoritmos e programação: foco nas competências de autoavaliação. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2012. v. 1, n. 1.
- VAHLDICK, A. et al. Pensamento computacional praticado com um jogo casual sério no ensino superior. p. 2303–2312, 2016.

WING, J. M. Computational thinking. *Communications of the ACM*, v. 49, n. 3, p. 33–35, 2006.

ZANETTI, H.; BORGES, M.; RICARTE, I. Pensamento computacional no ensino de programação: uma revisão sistemática da literatura brasileira. v. 27, n. 1, p. 21, 2016.

ZANINI, A. S.; RAABE, A. L. A. Análise dos enunciados utilizados nos problemas de programação introdutória em cursos de ciência da computação no Brasil. 2012.

ZORZO, A. et al. Referenciais de formação para os cursos de graduação em computação. sociedade brasileira de computação (sbc). 153p, 2017. 2017.

# A Questionário dos alunos AP1

Aqui estão as perguntas do questionário distribuído aos alunos do perfil 1, que cursaram a disciplina de Pensamento Computacional antes de cursarem Programação. As questões que possuem alternativas enumeradas são de múltipla escolha, enquanto as que possuem uma lista são de escolha única, e as que não possuem nenhuma alternativa são questões abertas para o participante responder com o texto que quiser.

1. Por que você escolheu cursar Licenciatura em Computação?
  - a) Desejo pela licenciatura
  - b) Aptidão e gosto pela computação
  - c) Cursar uma graduação em um horário que se adequa a minha realidade
  - d) Não consegui ingressar em outros cursos da área
  - e) Outros
  
2. Você mantém dupla jornada durante o curso, ou seja, estuda e trabalha em horários distintos?
  - Sim
  - Não
  
3. Se você mantém uma dupla jornada durante o curso, acredita que pode atrapalhar na sua dedicação ao aprendizado em programação?
  - Sim
  - Não
  
4. Você já considerou evadir ou já evadiu do curso (pensou em abandonar ou já abandonou o curso, tendo que ser reintegrado à graduação)?
  - Sim
  - Não
  
5. Se você foi reintegrado à graduação, qual ou quais foram os principais motivos para sua evasão?
  - a) Problemas em conciliar os estudos com a vida profissional
  - b) Tive sérias dificuldades com programação

- c) Dificuldades em outras disciplinas durante o curso que me deixaram desmotivado(a)
  - d) Problemas de saúde comigo ou algum parente próximo
  - e) Vim de outra área e tive dificuldade de me adaptar à área de computação
  - f) Outros
6. Você tinha algum tipo de receio em cursar Programação antes de entrar na graduação?
- Sim
  - Não
7. Se tinha algum receio, saber que teria que cursar uma disciplina introdutória, com o objetivo de preparar melhor os alunos aos conceitos mais básicos e fundamentais da programação, o(a) deixou mais tranquilo(a)?
- Sim
  - Não
8. Você acredita que o Pensamento Computacional te deixou mais preparado e confiante para aprender Programação?
- Sim
  - Não
9. Em um mundo onde cada vez mais estamos envolvidos com problemas e cenários interdisciplinares, que requerem uma gama de conhecimento em áreas diversas para solucioná-los, você acredita que o Pensamento Computacional contribui para isso?
- Sim
  - Não
10. Como você avalia seu conhecimento sobre raciocínio lógico antes de entrar na graduação?
- Nenhum
  - Ruim
  - Regular
  - Bom
  - Muito bom

11. Como você avalia seu conhecimento sobre pensamento algorítmico antes de entrar na graduação?
  - Nenhum
  - Ruim
  - Regular
  - Bom
  - Muito bom
  
12. Um dos propósitos de cursar Pensamento Computacional é auxiliar os discentes a desenvolver ou melhorar seu raciocínio lógico e algorítmico. Isso aconteceu com você?
  - Sim
  - Não
  
13. Sinta-se livre para deixar algum comentário mais específico sobre a sua experiência com o Pensamento Computacional, como ele o(a) ajudou a entender os objetivos do curso, como ele se inseriu dentro do seu contexto de vida ou alguma sugestão para as próximas turmas.

## B Questionário dos alunos AP2

Aqui estão as perguntas do questionário distribuído aos alunos do perfil 2, que cursaram a disciplina de Pensamento Computacional depois de cursarem Programação. As questões que possuem alternativas enumeradas são de múltipla escolha, enquanto as que possuem uma lista são de escolha única, e as que não possuem nenhuma alternativa são questões abertas para o participante responder com o texto que quiser.

1. Por que você escolheu cursar Licenciatura em Computação?
  - a) Desejo pela licenciatura
  - b) Aptidão e gosto pela computação
  - c) Cursar uma graduação em um horário que se adequa a minha realidade
  - d) Não consegui ingressar em outros cursos da área
  - e) Outros
  
2. Você mantém dupla jornada durante o curso, ou seja, estuda e trabalha em horários distintos?
  - Sim
  - Não
  
3. Se você mantém uma dupla jornada durante o curso, acredita que pode atrapalhar na sua dedicação ao aprendizado em programação?
  - Sim
  - Não
  
4. Você já considerou evadir ou já evadiu do curso (pensou em abandonar ou já abandonou o curso, tendo que ser reintegrado à graduação)
  - Sim
  - Não
  
5. Se você foi reintegrado à graduação, qual ou quais foram os principais motivos para sua evasão?
  - a) Problemas em conciliar os estudos com a vida profissional
  - b) Tive sérias dificuldades com programação

- c) Dificuldades em outras disciplinas durante o curso que me deixaram desmotivado(a)
  - d) Problemas de saúde comigo ou algum parente próximo
  - e) Vim de outra área e tive dificuldade de me adaptar à área de computação
  - f) Outros
6. Você tinha algum tipo de receio em cursar Programação antes de entrar na graduação?
- Sim
  - Não
7. Se tinha algum receio, desejaria ter cursado uma disciplina introdutória, com o objetivo de preparar melhor os alunos aos conceitos mais básicos e fundamentais da programação?
- Sim
  - Não
8. Você acredita que o Pensamento Computacional te deixaria mais preparado e confiante para aprender Programação, caso você tivesse cursado a disciplina de Pensamento Computacional antes de Programação?
- Sim
  - Não
9. Em um mundo onde cada vez mais estamos envolvidos com problemas e cenários interdisciplinares, que requerem uma gama de conhecimento em áreas diversas para solucioná-los, você acredita que o Pensamento Computacional contribui para isso?
- Sim
  - Não
10. Como você avalia seu conhecimento sobre raciocínio lógico antes de entrar na graduação?
- Nenhum
  - Ruim
  - Regular
  - Bom



- Muito bom

11. Como você avalia seu conhecimento sobre pensamento algorítmico antes de entrar na graduação?

- Nenhum
- Ruim
- Regular
- Bom
- Muito bom

12. Um dos propósitos de cursar Pensamento Computacional é auxiliar os discentes a desenvolver ou melhorar seu raciocínio lógico e algorítmico. Isso aconteceu com você?

- Sim
- Não

13. Sinta-se livre para deixar algum comentário mais específico sobre a sua experiência com o Pensamento Computacional, como ele o(a) ajudou a entender os objetivos do curso, como ele se inseriu dentro do seu contexto de vida ou alguma sugestão para as próximas turmas.