



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS I - CAMPINA GRANDE
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM BACHARELADO EM COMPUTAÇÃO**

DANILO MEDEIROS DANTAS

**ANÁLISE DA CONFIABILIDADE DE UM INSTRUMENTO NA MENSURAÇÃO DE
HABILIDADES COGNITIVAS PREDITORAS EM NOVATOS EM PROGRAMAÇÃO**

CAMPINA GRANDE - PB

2022

DANILO MEDEIROS DANTAS

**ANÁLISE DA CONFIABILIDADE DE UM INSTRUMENTO NA MENSURAÇÃO DE
HABILIDADES COGNITIVAS PREDITORAS EM NOVATOS EM PROGRAMAÇÃO**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Computação do Departamento de Computação do Centro de Ciências e Tecnologia da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de bacharel em Computação.

Orientador: Dra. Kézia de Vasconcelos Oliveira Dantas

CAMPINA GRANDE - PB

2022

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

D192a Dantas, Danilo Medeiros.

Análise da confiabilidade de um instrumento na mensuração de habilidades cognitivas preditoras em novatos em programação [manuscrito] / Danilo Medeiros Dantas. - 2022.

58 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências e Tecnologia, 2022.

"Orientação : Profa. Dra. Kézia de Vasconcelos Oliveira Dantas, Coordenação do Curso de Computação - CCT."

1. Introdução a Programação. 2. Teoria de Resposta ao Item - TRI. 3. Raciocínio matemático. 4. Habilidades preditoras.

I. Título

21. ed. CDD 005.3

DANILO MEDEIROS DANTAS

ANÁLISE DA CONFIABILIDADE DE UM INSTRUMENTO NA MENSURAÇÃO DE HABILIDADES COGNITIVAS PREDITORAS EM NOVATOS EM PROGRAMAÇÃO

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Computação do Departamento de Computação do Centro de Ciências e Tecnologia da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de bacharel em Computação.

Trabalho aprovado em 23 de Março de 2022.

BANCA EXAMINADORA


Prof. Dra. Kézia de Vasconcelos Oliveira Dantas (DC - UEPB)
Orientador(a)


Prof. Dr. Wellington Candeia de Araujo (DC - UEPB)
Examinador(a)


Prof. MSc. Jucelio Soares dos Santos (UEPB)
Examinador(a)

Dedico este trabalho a minha querida avó Eunice (in memoriam), por ser a mulher mais batalhadora e incrível do mundo que sempre cuidou de mim. Agradeço ao meu tio Antônio Eleno, que me deu todo o suporte para eu chegar até aqui. Sou grato também aos meus pais Osimar e Jeanne, que sempre me deram apoio. Sou grato a minha irmã, que sempre pude contar com ela. Agradeço aos meus amigos, que sempre estiveram comigo. Obrigado a minha namorada Renata por me estimular e estar comigo nos últimos anos. Obrigado meu primo Antônio Marcus que nunca me esqueceu e sempre apareceu para me ajudar e por fim, sou grato a todos os meus familiares que de alguma forma me ajudaram sempre que podiam.

AGRADECIMENTOS

Sou grato ao meu professor Jucelio, por todo o apoio e que fez tornar realidade este trabalho. Agradeço a minha orientadora Kézia, pela contribuição e apoio deste trabalho.

Agradeço a SPLAB, a Cognite, a CNPq, ao departamento de computação da UEPB e ao departamento de computação da UFCG pelas condições de logística para realização deste estudo.

Aos meus amigos de Bacharelado de computação, com os quais convivi durante toda a minha jornada de curso.

Aos professores e funcionários da UEPB, que fizeram parte de toda a minha trajetória.

*“Não podemos ensinar a outra pessoa diretamente;
só podemos facilitar sua aprendizagem.”*

Carl Rogers

RESUMO

Este trabalho promove reflexões sobre os desafios, propostas e possíveis soluções que envolvem as dificuldades de ingressantes em cursos Técnicos e Superiores nas disciplinas Introdutórias de Programação (CS1). O intuito é melhorar a capacidade de triagem de novatos em programação para identificar sinais de falhas e evasão antes mesmo deles iniciarem um CS1. Para tanto, identificamos as habilidades preditoras de programação; desenvolvemos itens que contemplem estes indicadores; e por fim, calibramos o banco de itens por meio da Teoria de Resposta ao Item em termos de informações que os itens fornecem a respeito do constructo psicológico específico avaliado. A principal contribuição desta pesquisa foi a construção de um instrumento com itens que permitissem levantar indícios de falhas em um CS1, proporcionando assim, o professor traçar estratégias que contribuam para o desenvolvimento do discente previamente. Como resultados, os itens que desenvolvemos apresentam uma boa análise de conteúdo, boas propriedades psicométricas e os instrumentos apresentam boa confiabilidade em avaliações profissionais.

Palavras-chaves: Triagem. Habilidades Preditoras de Programação. Mensuração. Teoria de Resposta ao Item.

ABSTRACT

This work promotes reflections on the challenges, proposals, and possible solutions that involve the difficulties of newcomers to Technical and Higher Education courses in Introductory Programming disciplines (CS1). The intent is to improve the screening capability of programming novices to identify signs of failure and evasion even before they start a CS1. To do so, we identify predictive programming skills, develop items that address these indicators, and finally, we calibrate the item bank through Item Response Theory in terms of information that the items provide about the specific psychological construct evaluated. The main contribution of this research was the construction of an instrument with items that allowed to raise evidence of failures in CS1, thus providing the teacher to trace strategies that contributed to the development of the student previously. As a result, the items we developed present an exemplary content analysis with good psychometric properties, and the instruments present good reliability in professional assessments.

Keywords: Screening. Predictive Programming Skills. Measurement. Item Response Theory.

LISTA DE ILUSTRAÇÕES

Figura 1 – CCI's exemplo	24
Figura 2 – FII's exemplo	26
Figura 3 – Exemplo de Item da Habilidade Resolução de Problemas	39
Figura 4 – Exemplo de Item da Habilidade Flexibilidade Cognitiva	39
Figura 5 – Exemplo de Item da Habilidade Raciocínio Matemático	40
Figura 6 – Exemplo de Item da Habilidade Flexibilidade Cognitiva	40
Figura 7 – CCI's e FII's dos itens da Habilidade Resolução de Problemas.	43
Figura 8 – CCI's e FII's dos itens da Habilidade Pensamento Abstrato.	45
Figura 9 – CCI's e FII's dos itens da Habilidade Raciocínio Matemático.	47
Figura 10 – CCI's e FII's dos itens da Habilidade Flexibilidade Cognitiva.	49

LISTA DE TABELAS

Tabela 1 – Análise de Confiabilidade entre Juízes	38
Tabela 2 – Consistência Interna do Instrumento	41
Tabela 3 – Habilidade Resolução de Problemas - Todos os Itens Calibrados	42
Tabela 4 – Habilidade Resolução de Problemas - Todos os Itens Ajustados	42
Tabela 5 – Habilidade Pensamento Abstrato - Todos os Itens Calibrados	44
Tabela 6 – Habilidade Pensamento Abstrato - Todos os Itens Ajustados	44
Tabela 7 – Habilidade Raciocínio Matemático - Todos os Itens Calibrados	46
Tabela 8 – Habilidade Raciocínio Matemático - Todos os Itens Ajustados	46
Tabela 9 – Habilidade Flexibilidade Cognitiva - Todos os Itens Calibrados	48
Tabela 10 – Habilidade Flexibilidade Cognitiva - Todos os Itens Ajustados	48

LISTA DE QUADROS

Quadro 1 – Detalhes da pesquisa e seleção de estudos por banco de dados	31
Quadro 2 – Detalhes do estudo da fase de pesquisa	31
Quadro 3 – Habilidades preditoras de programação	36
Quadro 4 – Instrumentos que medem habilidades preditoras de programação	38

LISTA DE ABREVIATURAS E SIGLAS

3ML	Modelo Logístico Unidimensional de 3 Parâmetros
CCI	Curva Característica do Item
CS1	Curso Introdutório de Programação
EAP	Expected a Posterior
FII	Função de Informação do Item
QP	Questão de Pesquisa
RSL	Revisão Sistemática da Literatura
TCT	Teoria Clássica dos Testes
TRI	Teoria de Resposta ao Item

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Cenário Técnico Científico	14
1.2	Problema	15
1.3	Objetivos	16
1.4	Questões de Pesquisa	16
1.5	Estrutura do trabalho	17
2	TEMAS E TRABALHOS RELACIONADOS	18
2.1	Habilidades Predictoras de Programação	18
<i>2.1.1</i>	<i>Resolução de Problemas</i>	<i>18</i>
<i>2.1.2</i>	<i>Pensamento Abstrato</i>	<i>19</i>
<i>2.1.3</i>	<i>Raciocínio Matemático</i>	<i>19</i>
<i>2.1.4</i>	<i>Flexibilidade Cognitiva</i>	<i>20</i>
2.2	Teorias Psicométricas	20
<i>2.2.1</i>	<i>Teoria Clássica dos Testes</i>	<i>20</i>
<i>2.2.1.1</i>	<i>Coeficiente de Ponto Bisserial</i>	<i>21</i>
<i>2.2.1.2</i>	<i>Coeficiente Alpha de Cronbach</i>	<i>21</i>
<i>2.2.1.3</i>	<i>Limitações da TCT</i>	<i>22</i>
<i>2.2.2</i>	<i>Teoria de Resposta ao Item</i>	<i>23</i>
<i>2.2.2.1</i>	<i>Curva Característica do Item</i>	<i>23</i>
<i>2.2.2.2</i>	<i>Estimativa da Habilidade</i>	<i>25</i>
<i>2.2.2.3</i>	<i>Função de Informação do Item</i>	<i>25</i>
3	METODOLOGIA	28
3.1	Revisão Sistemática de Literatura	28
<i>3.1.1</i>	<i>Objetivo e escopo</i>	<i>29</i>
<i>3.1.2</i>	<i>Estratégia de Pesquisa</i>	<i>29</i>
<i>3.1.3</i>	<i>Estratégia de Seleção de Estudos</i>	<i>29</i>
<i>3.1.4</i>	<i>Extração dos Dados</i>	<i>30</i>
3.2	Construção do Banco de Itens	31
3.3	Calibração do Banco de Itens	32
4	ANÁLISE E RESULTADOS	35
4.1	Revisão Sistemática de Literatura	35
<i>4.1.1</i>	<i>Identificação das Habilidades Predictoras de Programação</i>	<i>35</i>
<i>4.1.2</i>	<i>Mensuração das Habilidades Predictoras de Programação</i>	<i>37</i>
4.2	Construção do Banco	37
4.3	Calibração do Banco de Itens	41

4.3.1	<i>Resolução de Problemas</i>	42
4.3.2	<i>Pensamento Abstrato</i>	44
4.3.3	<i>Raciocínio Matemático</i>	46
4.3.4	<i>Flexibilidade Cognitiva</i>	48
5	CONSIDERAÇÕES FINAIS E SUGESTÕES PARA TRABALHOS FUTUROS	50
	REFERÊNCIAS	51
	ANEXO A – TERMO DE ASSENTIMENTO	57
	ANEXO B – TERMO DE CONSENTIMENTO LIVRE ESCLARECIDO	58

1 INTRODUÇÃO

Neste capítulo, apresentamos uma visão geral desta pesquisa, de modo a descrever o cenário técnico científico, problema, objetivos e questões de pesquisas.

1.1 Cenário Técnico Científico

Geralmente os cursos de Ensino Superior na área da Computação, presenciais ou não, apresentam várias disciplinas de programação como foco central. Uma delas aborda conhecimentos introdutórios de programação. Essa disciplina é conhecida nos Estados Unidos da América como CS1, trata-se de uma disciplina desafiadora para novatos que convencionalmente cobre habilidades de resolução de problemas, conceitos básicos de programação, sintaxe e semântica de uma linguagem de programação para formular soluções (ALSHAYE; TASIR; JUMAAT, 2019).

Nos últimos anos, o CS1 vem apresentando altos índices de evasão e reprovação, equivalente a um terço dos estudantes matriculados nesta disciplina no mundo todo (WATSON; LI, 2014). Tal fato gera desmotivação nos alunos e desconfiança na comunidade acadêmica quanto à qualidade de tais cursos superiores (SILVA; SILVA; SANTOS, 2009). Um dos fatores que podem influenciar esses indicadores negativos é a adoção do modelo de ensino tradicional (BORGES, 2000).

Tradicionalmente, o ensino de programação é de difícil assimilação, com base em lições teóricas, apresentação de exemplos de pseudocódigo e posteriormente resolução de problemas. O professor se concentra em uma série de comandos e códigos de uma linguagem de programação específica, selecionando exercícios como exemplo, geralmente apresentando-os em um contexto não prático, o que dificulta a criação de imagens e modelos mentais que ajudam na construção dos algoritmos (RAPKIEWICZ et al., 2007). Os alunos que estão aprendendo a programar precisam aprender a pensar como um programador, o que envolve fazer planos necessários para resolver problemas, e devem aprender a escrever o código que corresponde aos planos sobre os quais pensaram. Ser um bom programador significa ser criativo e ter uma invenção. Conhecimento específico em programação pode ser aprendido com base nas necessidades de cada programador (TAHERI; SASAKI; NGETHA, 2015).

No entanto, os alunos entram na sala de aula introdutória com uma ampla gama de níveis de habilidade, particularmente seu conhecimento prévio em programação e habilidade de resolução de problemas (SANDS, 2019). Na primeira fase da aprendizagem de programação, os alunos com baixo desempenho acham difícil e não serão inspirados a programar (TAHERI; SASAKI; NGETHA, 2015). Eles costumam ter dificuldades durante a análise de problemas, planejamento e soluções de design (SMETSERS; SMETSERS, 2017).

Eles têm dificuldade em analisar o fluxo de um programa, não conseguem corrigir erros ou não conseguem depurar o programa, geralmente dependendo de outros para solução de problemas e correção de erros. A atividade mais importante para os alunos enquanto aprendem

a programar é aprender como criar seus programas. Portanto, os alunos escrevem programas depois de aprender as regras de sintaxe e alguns exemplos. No entanto, mais ênfase precisa ser colocada nas habilidades precursoras para escrever código.

Mesmo depois de passar por CS1, estudos mostram que os alunos ainda têm sérios problemas na aplicação dos conceitos em programação, considerando-a uma tarefa complexa (MCCRACKEN et al., 2001). Estudos também apontam que a maioria dos alunos tem seu conhecimento de programação consolidado apenas no final da segunda disciplina de programação (TEW; MCCRACKEN; GUZDIAL, 2005). Por esta razão, ensinar programação é considerada um dos grandes desafios da área de Educação na Computação (CASPERSEN; CHRISTENSEN, 2000)

Para lidar com essas diferenças, os educadores de informática devem ajustar o design instrucional do ensino tradicional e escolher tarefas menos densas para diminuir a carga cognitiva dos alunos (CABO; LANSIQUOT, 2014). Apesar dos avanços nos métodos, processos, abordagens e instrumentos para o ensino e aprendizagem da programação introdutória, as taxas de evasão e reprovação ainda são altas em CS1 (SANDS, 2019).

1.2 Problema

Não há consenso sobre como as instituições devem trabalhar as habilidades prévias em um CS1. Até onde sabemos, nenhum estudo apresentou uma proposta de solução para melhorar a capacidade de triagem de novatos em programação que forneçam, por exemplo, instrumentos confiáveis para identificar sinais de falhas e evasão de alunos no CS1. Os estudos existentes abordam os desafios de ensino/avaliação em CS1, métodos alternativos, feedback formativo (LUXTON-REILLY et al., 2018). Ou mencionam algumas habilidades de programação sem classificação ou hierarquia cognitiva (MEDEIROS; RAMALHO; FALCÃO, 2018). Esses estudos são essenciais porque fornecem uma visão geral dos fundamentos em CS1 e indicam as habilidades que devem ser promovidas e avaliadas, mas não impedem quais habilidades devem ser trabalhadas em um CS1 no nível preditivo ou introdutório.

Ambos os trabalhos caracterizaram os desafios em CS1. Destacando algumas habilidades fundamentais para um aluno iniciante aprender a programar e as dificuldades que enfrentou nesse processo. O principal desafio do ensino diz respeito à falta de métodos e ferramentas adequados para o ensino personalizado. A resolução de problemas continua a ser um dos principais desafios de aprendizagem, seguida de motivação e envolvimento e dificuldade em aprender a sintaxe das linguagens de programação.

Portanto, há uma necessidade de uma pesquisa que forneça uma melhor compreensão sobre as habilidades prévias à programação e destaque estratégias para mensurar tais habilidades, permitindo que os professores possam diagnosticar/rastrear precocemente falhas de novatos em CS1.

1.3 Objetivos

A presente pesquisa teve como objetivo principal melhorar a capacidade de triagem de novatos em programação, fornecendo instrumentos confiáveis para identificar sinais de falhas e evasão de alunos em disciplinas introdutórias de programação¹. Para alcançar o objetivo geral desta pesquisa, foram necessários atingir os seguintes objetivos específicos:

- Identificar as habilidades preditoras de programação;
- Identificar métricas para mensurar as habilidades preditoras de programação;
- Desenvolver itens com boa análise de conteúdo que contemplem os indicadores de habilidades preditoras de programação;
- Calibrar o banco de itens em termos das informações que fornecem a respeito do constructo psicológico específico avaliado;
- Avaliar a confiabilidade dos instrumentos por meio das teorias de mensuração.

1.4 Questões de Pesquisa

A principal questão de pesquisa deste estudo é: o instrumento baseado na Teoria de Mensuração melhora a capacidade de triagem de novatos em programação para identificar sinais de falhas e evasão antes mesmo deles iniciarem um CS1? A fim de entender melhor se é possível identificar se um novato em programação tende a falhar/evadir ou não de um CS1, esta pesquisa se concentra nas seguintes Questões de Pesquisa (QP) dentro do escopo mais amplo da avaliação de habilidades prévias de programação:

- **QP1.** Quais são as habilidades preditoras de programação?
- **QP2.** Como devemos mensurar habilidades preditoras de programação?
- **QP3.** Os itens que contemplem os indicadores de habilidades preditoras de programação possuem boa análise de conteúdo?
- **QP4.** Os itens que contemplem os indicadores de habilidades preditoras de programação possuem boas propriedades psicométricas?
- **QP5.** Os instrumentos desenvolvidos a partir do banco de itens apresentam boa confiabilidade?

¹ Ressaltamos que esse Trabalho de Conclusão de Curso foi fruto de parceria de uma pesquisa doutoral desenvolvida na UFCG, junto ao Programa de Pós-Graduação em Ciência da Computação, com o Prof. Me. Jucelio Soares dos Santos e seus orientadores Prof. Dr. Wilkerson de Lucena Andrade e Prof. João Arthur Brunet Monteiro como autor da tese.

1.5 Estrutura do trabalho

Este trabalho apresenta cinco capítulos e está organizado da seguinte maneira: no Capítulo 1, apresentamos uma visão geral desta investigação com relação ao cenário técnico científico, problema, objetivos e questões de pesquisa; no Capítulo 2, apresentamos os temas e trabalhos relacionados à pesquisa; no Capítulo 3, apresentamos a metodologia desta pesquisa; no Capítulo 4, apresentamos e discutimos os resultados; no Capítulo 5, apresentamos as considerações finais e sugestões para trabalhos futuros; e ao final, encontra-se as referências e os apêndices utilizados do decorrer desta pesquisa.

2 TEMAS E TRABALHOS RELACIONADOS

Neste capítulo, apresentamos o embasamento teórico a partir de várias áreas e trabalhos que se relacionam e que permitem caracterizar esta pesquisa.

2.1 Habilidades Predictoras de Programação

Promoveremos nesta seção, reflexões sobre os desafios, propostas e possíveis soluções, dos últimos 10 anos, que envolvem as habilidades predictoras de programação em cursos Técnicos e Superiores em CS1.

2.1.1 Resolução de Problemas

Resolução de problemas é um conceito útil que descreve o esforço consciente no processamento controlado da informação que identifica, descobre ou inventa uma solução para um problema (ALSHAYE; TASIR; JUMAAT, 2019). A solução de problemas requer muito mais do que ter um modelo mental válido de construções básicas. Transmitir tais habilidades genéricas de resolução de problemas requer um modelo construtivista mais orientado para projetos, onde os alunos projetam e constroem artefatos (THEVATHAYAN; HAMILTON, 2015).

A resolução de problemas desempenha um papel significativo nas capacidades técnicas das pessoas em geral, ainda mais para estudantes de Ciências Aplicadas e Engenharia, como Ciência da Computação (CHAUDHRY; RASOOL, 2012). Vários campos usam a resolução de problemas como uma habilidade essencial para o desenvolvimento profissional; portanto, é útil ter alguma familiaridade básica com a resolução de problemas, independentemente do trabalho ou estudo (TAHERI; SASAKI; NGETHA, 2015).

Habilidades de resolução de problemas são consideradas parte integrante da compreensão dos conceitos precisos do domínio de programação para iniciantes em CS1. No entanto, essas habilidades são principalmente abordadas apenas nas primeiras aulas de tais cursos ou incluídas em apenas alguns capítulos de abertura de alguns livros relevantes (MATHEW; MALIK; TAWAFK, 2019). Infelizmente, os currículos tradicionais da maioria das universidades não ensinam essa habilidade no nível desejado pelos empregadores (CHAUDHRY; RASOOL, 2012). Como resultado, CS1 têm altas taxas de reprovação e evasão (MATHEW; MALIK; TAWAFK, 2019). Outras pesquisas neste campo revelaram que a falta de habilidade para resolver problemas é considerada uma das principais deficiências enfrentadas pelos novatos, e exacerbada pela sintaxe da linguagem que os novatos usam (HOOSHYAR et al., 2015).

Habilidades de resolução de problemas são medidas e estimuladas com exercícios e treinamento (CHAUDHRY; RASOOL, 2012). Dessa forma, os alunos com habilidades de resolução de problemas mais eficazes acham a programação fácil e podem dominar a programação com pouca ou nenhuma dificuldade, independentemente do ambiente de programação. Ao contrário, os alunos com baixa habilidade de resolução de problemas acham a programação difícil de

entender e muitas vezes são incapazes de dominá-la (MLADENOVÍĆ; KRPAŃ; MLADENOVÍĆ, 2017). A habilidade de resolução de problemas produz um benefício real e quantificável para a habilidade de programação dos alunos (KOULOURI; LAURIA; MACREDIE, 2014). Além disso, ter um conhecimento útil de técnicas de resolução de problemas é vital (TAHERI; SASAKI; NGETHA, 2015). Porque a capacidade de resolver problemas se correlaciona significativamente com o desempenho em atribuições de programação (LISHINSKI et al., 2016).

As habilidades de programação envolvem não apenas a resolução de problemas, mas também um estilo de pensamento. Um estilo de pensamento é uma forma preferida de pensar, seja global (abstrato) ou local (concreto) (LÓPEZ et al., 2017). É uma estratégia essencial para o desenvolvimento de programas de computador, pois o aluno precisa testar e avaliar suas soluções de programação de computador. Para testar e avaliar adequadamente essas soluções, os alunos precisam de boas habilidades metacognitivas. Em outras palavras, os alunos precisam ser capazes de "pensar sobre pensar", ou seja, pensar de forma abstrata.

2.1.2 *Pensamento Abstrato*

O pensamento abstrato influencia a maneira como as pessoas aprendem informações, formam julgamentos e regulam o comportamento. Por exemplo, em estruturas de iteração, o pensamento abstrato pode afetar diretamente a compreensão de um programa que usa a estrutura de iteração. Se o nível de pensamento abstrato do aluno for mais alto, ele compreenderá melhor o programa com uma estrutura de iteração. Quando o professor fornece ao aluno um problema de programação descomplicado, a habilidade de pensamento abstrato não é discriminatória na avaliação dos alunos. Porém, quando um problema de programação é difícil, o alto nível de capacidade de pensamento abstrato desempenha um papel essencial na solução do problema (PARK; HYUN, 2014).

Existe um efeito moderador na habilidade de programar entre o nível de familiaridade com programação e pensamento abstrato (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015). Quando os alunos têm um alto nível de habilidades de pensamento abstrato, suas habilidades de programação melhoram muito mais do que aqueles com um baixo nível de habilidades de pensamento abstrato. Quando os professores aumentam o nível de pensamento abstrato dos alunos, a distância psicológica temporal também aumenta (SKALKA; DRLÍK, 2018). Além disso, a familiaridade é um fator crucial para a compreensão do programa; os alunos aprendem mais e praticam mais sobre a linguagem de programação no futuro (STATTER; ARMONI, 2020).

2.1.3 *Raciocínio Matemático*

Habilidades de programação são frequentemente associadas às habilidades de resolução de problemas, e certos tipos de habilidades matemáticas afetam as habilidades analíticas que contribuem para a experiência de aprender a programar no computador. Além da sintaxe e semântica de uma linguagem de programação, pensamento lógico e matemático, habilidades de numeração e visualização, conhecimento de álgebra e cálculo, conhecimento de funções

condicionais e recursivas são essenciais para projetar funções e procedimentos durante a atividade de programação (ATTALLAH; ILAGURE; CHANG, 2018).

Para escrever um código totalmente especificado e verificável, os alunos de computação precisam ser capazes de raciocinar matematicamente sobre os componentes de software e seus relacionamentos, precisam modelá-los por meio de construções matemáticas e precisam entender e escrever especificações e afirmações formais usando a linguagem precisa da notação matemática (DRACHOVA et al., 2015). Existe uma correlação entre habilidades matemáticas e potencial de programação. Os alunos consideram a lógica da programação e da metodologia de ensino como as principais fontes de dificuldades (SOUZA et al., 2019). Conceitos matemáticos relacionados à indução, inferência e raciocínio lógico estão entre eles.

2.1.4 Flexibilidade Cognitiva

Recentemente, um estudo mostra que a flexibilidade cognitiva é um indicador significativo do desempenho acadêmico no treinamento em CS1. A flexibilidade cognitiva é uma das competências individuais mais preliminares que podem orientar o aluno durante a regulação e avaliação da solução e a saída no processo de resolução de problemas (DURAK, 2018). Essa habilidade está ligada ao processo de aprendizagem, mostrando orientação, aconselhamento e verificação.

A flexibilidade cognitiva permite ao aluno empregar as estratégias de aprendizagem mais eficazes relacionadas ao tema em estudo ou identificar as etapas para resolver um problema, encontrar uma solução, controlar o processo de aprendizagem e controlar os produtos ou oportunidades de autorregulação. Desta forma, os professores devem desenvolver esta habilidade a fim de melhorar o desempenho dos alunos em programação (DURAK, 2020).

2.2 Teorias Psicométricas

Como nosso objetivo principal é analisar alguma característica não observável, seja um conhecimento em uma determinada disciplina na área educacional, seja uma patologia ou comportamento na área da saúde. Essa característica não-observável é denominada de traço latente ou habilidade (PRIMI, 2012). Existem duas teorias que são utilizadas para este fim: a Teoria Clássica dos Testes (TCT) e a Teoria de Resposta ao Item (TRI), ambas são usadas em conjunto para analisar as informações do instrumento por meio da estimativa de parâmetros dos itens e a análise do instrumento de medida como um todo.

2.2.1 Teoria Clássica dos Testes

A TCT considera a pontuação total de um instrumento como a medida primária para medir as habilidades ou conceitos de um indivíduo. A TCT usa padrões para interpretar a pontuação de um instrumento. Essas normas comparam as pontuações de dois indivíduos ou situam a posição de um indivíduo no construto medido em um determinado instrumento (ARAÚJO et al., 2019).

A TCT está associada à diferença entre a pontuação do indivíduo não testado e o valor real dessa pontuação (PRIMI, 2012). A TCT se propõe a verificar se o construto mede ou deve medir (PASQUALI, 2017). Por meio do escore, é possível utilizar algumas medidas para avaliar a qualidade dos itens e do instrumento, como o coeficiente de correlação do ponto bisserial e o coeficiente alfa de Cronbach.

2.2.1.1 Coeficiente de Ponto Bisserial

É possível calcular o coeficiente de correlação entre duas variáveis em testes tradicionais, uma numérica e outra nominal categórica. Nesse caso, a variável categórica possui apenas dois valores possíveis (certo/errado); uma dessas variáveis é chamada de dicotômica. Em seguida, para calcular a correlação entre esta variável e outra variável (numérica), proceda-se com o coeficiente de Pearson da forma usual, dada a normalidade da amostra, denominado coeficiente de correlação de ponto bisserial (BAKER, 2001).

O coeficiente do ponto bisserial é a correlação de Pearson entre as variáveis dicotômicas e a pontuação do instrumento. É usado como uma medida para discriminar o item ao resultado de um instrumento e indica a capacidade do item em diferenciar pessoas com habilidade forte e fraca na tarefa testada (ARAÚJO et al., 2019). A Equação 2.1 define o coeficiente do ponto bisserial:

$$Ppb = \frac{\bar{X}_A - \bar{X}_T}{S_T} \sqrt{\frac{p}{1-p}} \quad (2.1)$$

Onde,

\bar{X}_A representa a média global das pontuações dos respondentes que acertaram o item;

\bar{X}_T representa a média global das pontuações do instrumento;

S_T representa o desvio padrão do instrumento;

p representa a proporção de respondentes que acertaram o item.

O coeficiente de ponto bisserial estima os itens que têm o maior impacto na habilidade estimada; se o avaliado acertar esse item, terá boas chances de passar no exame. O coeficiente do ponto bisserial varia entre -1 e +1, e quanto mais próximo de 1 mais discriminante é o item, quanto maior o valor do coeficiente, mais forte é a correlação deste item com a pontuação total. Este valor mostrará que aquele item é essencial para o instrumento. Por exemplo, se o item 4 tem um coeficiente de 0,84 e o item 5 tem um coeficiente de 0,43, significa que o item 4 é mais discriminativo do que o item 5. Indivíduos que acertam o item 4 têm maior probabilidade de obter um melhor resultado em um instrumento.

2.2.1.2 Coeficiente Alpha de Cronbach

Para construir um instrumento é necessário verificar a sua confiabilidade. A consistência interna mede a confiabilidade do instrumento, ou seja, examinar a homogeneidade dos itens que compõem o instrumento, verificando a magnitude das relações entre os itens e o escore

total. Podemos calcular a consistência interna a partir da pontuação geral do instrumento e da pontuação de cada item. A Equação 2.2 define o coeficiente de alfa de Cronbach:

$$\alpha = \frac{n}{n-1} \left(1 - \frac{\sum S_i^2}{S_T^2} \right) \quad (2.2)$$

Onde,

n representa o número dos itens;

$\sum S_i^2$ representa a soma das variâncias para n itens;

S_T^2 representa o intervalo global de pontuações de teste.

Calculamos a consistência usando o coeficiente alfa de Cronbach variando de 0 a 1; quanto mais próximo de 1 indica que o Instrumento possui consistência interna adequada (ANDRADE; LAROS; GOUVEIA, 2010). Valores próximos de 1 indicam que o Instrumento possui consistência interna adequada. Valores entre 0,70 e 0,80 são considerados aceitáveis, mas com ressalvas. Quando os valores estão abaixo de 0,70, os itens que compõem o instrumento precisam ser reavaliados pelo pesquisador (ARAÚJO et al., 2019).

2.2.1.3 Limitações da TCT

A TCT seleciona analiticamente os melhores itens de um banco de itens, evitando redundâncias e facilitando o processo de mensuração da habilidade. Entre as medidas desta teoria, tem duas características primordiais dos itens, são elas: a dificuldade (obtido pela proporção de indivíduos que respondem afirmativamente a esse item) e a discriminação (distingue o grupo de indivíduos que obtiveram alta pontuação dos que obtiveram baixa pontuação no teste, através da diferença entre essas pontuações).

O índice de dificuldade varia entre 0 (significa que nenhum indivíduo respondeu de forma afirmativa ao item) e 1 (significa que todos indivíduos responderam afirmativamente). O parâmetro de discriminação pode variar entre -1 e +1. Os fatores podem influenciar a interpretação do índice de dificuldade do item: o número de itens não respondidos (por qualquer motivo que seja) e a probabilidade de um indivíduo responder o item corretamente ao acaso (chute). Desta forma, se um grande número de indivíduos não responder ao item por falta de tempo, o índice de dificuldade não será avaliado corretamente. Assim, acontecerá se um grande número de indivíduos responder corretamente de maneira aleatória (PRIMI, 2012).

Apesar de simples e direta, essa teoria apresenta limitações em relação ao número certos de itens a ser aplicado no instrumento, de tal forma que os indivíduos precisam responder a todos itens presentes no banco de itens, fornecendo respostas a itens que não conduzem a seu nível de habilidade, causando exaustão e gerando respostas não confiáveis. Por sua vez, esses respondentes estão sujeitos a fornecer erros durante a aplicação do instrumento, pois diferentes respondentes podem receber um conjunto de itens de tamanhos diferentes, resultando em uma medição menos precisa da proficiência. Também estão propícios a erros durante o processo de correção, que ocorre de forma dispendiosa e requer tempo do aplicador. Uma outra limitação da

TCT é o fato de que o instrumento de medida considera a individualização dos avaliados pelo instrumento (BAKER, 2001).

2.2.2 Teoria de Resposta ao Item

Tendo em vista as limitações provenientes da TCT, a TRI surge com a possibilidade de complementá-la. Uma vez que com a TCT verifica a possibilidade de o indivíduo acertar ao acaso determinado item, com a TRI essa possibilidade pode ser identificada por meio das respostas a outros itens. Com a TRI é possível também que se estabeleça uma comparação entre indivíduos de uma mesma população que foram submetidos a testes diferentes, ou ainda, a comparação entre populações por meio de testes com alguns itens em comum (ANDRADE; TAVARES; VALLE, 2000).

Podemos usar a TRI para elaborar instrumentos de avaliação educacional, calibração de itens (caracterização de itens por valores de parâmetros numéricos) e outros processos de desenvolvimento de instrumentos. Com a TRI é possível ajustar os dados para o modelo. Assim, pessoas diferentes ou a mesma pessoa podem ter suas habilidades comparadas aos itens do instrumento com frequência, pois usam parâmetros que são medidos estatisticamente, independentemente da amostra usada (BAKER, 2001).

Na TRI, um conjunto de fatores ou variáveis hipotéticas pode prever o comportamento do indivíduo em um item. Além disso, a dependência entre comportamento e a habilidade pode estar relacionada a uma função matemática monótona crescente, cujo gráfico é denominado Curva Característica do Item (CCI) (PASQUALI, 2017).

2.2.2.1 Curva Característica do Item

A CCI fornece informações sobre a probabilidade de cada um acertar o item (BAKER, 2001). Cada item, na TRI, possui uma curva característica que determina a sua qualidade. A CCI é o gráfico que representa a relação entre a habilidade estimada e o desempenho nos itens (ARAÚJO et al., 2019). A CCI é influenciada por particularidades da TRI, de acordo com o modelo matemático utilizado.

Diferentes modelos matemáticos podem ser usados dependendo do número de parâmetros envolvidos, dimensionalidade ou tipo de itens presentes no instrumento. Consideramos o modelo logístico unidimensional de três parâmetros (3ML): discriminação, dificuldade e probabilidade de acertar o item pelo acaso. A Equação 2.3 define o 3ML:

$$P(\theta) = c_j + (1 - c_j) \frac{1}{1 + e^{-a_j(\theta - b_j)}} \quad (2.3)$$

Onde,

θ representa o traço/habilidade latente de um indivíduo;

a_j representa o parâmetro de discriminação do item;

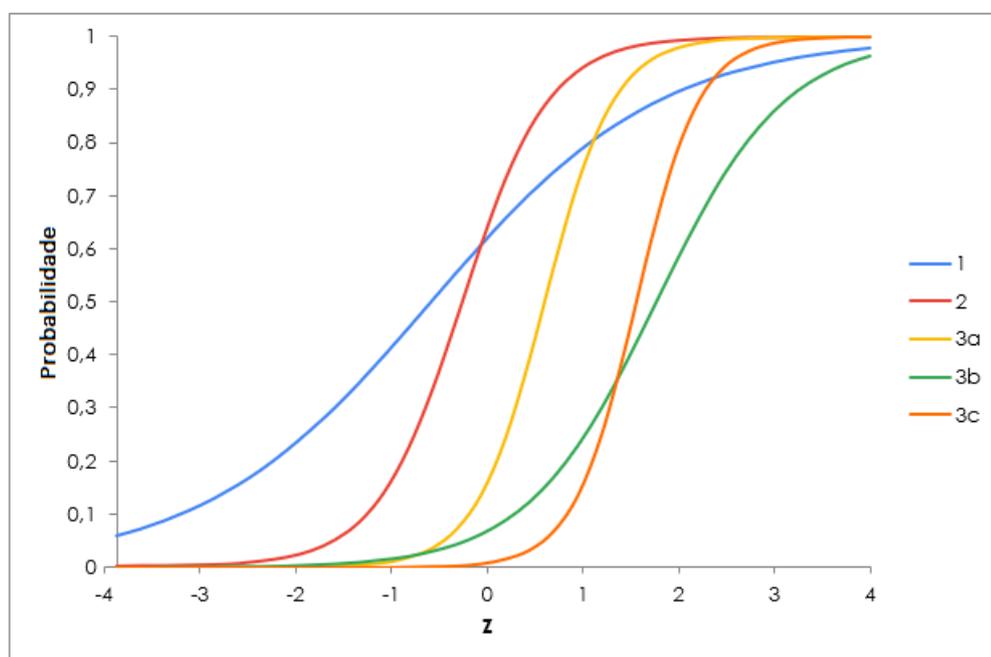
b_j representa o parâmetro de dificuldade do item;

c_j representa o parâmetro de acertar o item pelo acaso;

A discriminação do item (parâmetro a) descreve quantos indivíduos de diferentes habilidades distinguem-se quanto as chances de acertar o item, detalhando sujeitos com magnitudes próximas à habilidade avaliada. Este parâmetro varia entre 0 (nada discriminativo) até 4 (extremamente discriminativo). Por sua vez, a dificuldade do item (parâmetro b) refere-se à habilidade necessária para um indivíduo com uma dada probabilidade de acertar o item, calculada mediante a chance de acertar o item pelo acaso. Este parâmetro varia de -3 (itens fáceis) até +3 (itens difíceis). E, a probabilidade de acerto ao acaso (parâmetro c) refere-se à chance de um indivíduo com baixa habilidade acertar um item difícil, ou seja, acertar pelo chute. Este parâmetro varia entre 0 e 0,5 (ARAÚJO et al., 2019).

Por meio da multiplicidade de procedimentos de programas de computador especializados podemos obter os parâmetros do item. Esses programas usam funções matemáticas não lineares, como funções logarítmicas, que produzem CCI, e representações gráficas de funções matemáticas que relacionam a probabilidade de resposta do item ao nível de traço latente ou habilidade (ARAÚJO et al., 2019). A seguir, apresentamos na Figura 1 um exemplo de gráfico que plota algumas CCI's.

Figura 1 – CCI's exemplo



Fonte: Elaborada pelo autor (2022).

O gráfico da Figura 1 mostra cinco CCI's, cada um correspondendo a um item, modelado a partir da Equação 2.3, que usaremos como exemplos para ilustrar o comportamento dos parâmetros. O valor da habilidade Z pode assumir qualquer número real. Geralmente, o eixo x que representa a escala de habilidade Z é classificado de -4 a +4. O eixo y representa a probabilidade do item de resposta correta variando de 0 a 1. As linhas indicam o nível de

dificuldade quando a probabilidade de resposta correta é 50%, pois os parâmetros de dificuldade e habilidade Z são na mesma escala. No exemplo da Figura 1, o item ‘3c’ é o item mais discriminador, ‘3b’ é o mais difícil, e o item ‘1’ é o mais fácil e menos discriminatório.

2.2.2.2 Estimativa da Habilidade

Introduzido anteriormente, 3ML reproduz uma escala chamada traço ou habilidade latente. O nível gerado é padronizado (média = 0 e DP = 1) e, conforme observado para o parâmetro b métrico, em tese, essa escala pode variar de -4 a +4. Assim, os escores são estimados pela TRI usando um método de estimação (FOX; GLAS, 2001).

Neste estudo, usamos o Expected A Posteriori (EAP) (KOLEN; TONG, 2010) (LU; THOMAS; ZUMBO, 2005); cada pessoa recebe a pontuação que melhor identifica sua habilidade na escala. A EAP deriva de princípios estatísticos bayesianos. O termo “a posteriori” deriva do conceito bayesiano de probabilidade posterior. Neste contexto, refere-se a uma distribuição de probabilidade posterior de pontuações de traços latentes - especificamente, a distribuição prevista de pontuações para um determinado caso dada (a) a resposta padrão desse caso e (b) os parâmetros estimados do modelo. O termo “Expected” deriva do conceito de um valor esperado. Assim, uma estimativa “esperada a posteriori” refere-se ao valor esperado da distribuição de probabilidade posterior das pontuações do traço latente para um dado caso. A Equação 2.4 define a estimativa do traço latente pela EAP:

$$\theta_{s+1} = \theta_s + \frac{\sum_{i=1}^n a_i [u_i - P_i \theta_s]}{\sum_{i=1}^n a_i^2 P_i(\theta_s) Q_i(\theta_s)} \quad (2.4)$$

Onde,

θ_s representa a capacidade estimada do examinado dentro de s iterações;

a_i representa o parâmetro de inclinação do item i , $i = 1, 2, \dots, N$;

O procedimento EAP estima a capacidade de um examinador. EAP é uma média da distribuição posterior e do erro padrão após a aplicação do instrumento e depende da Função de Informação do Item (FII) e seus parâmetros.

2.2.2.3 Função de Informação do Item

FII analisa o quanto um item contém informações psicométricas para medir a habilidade. Sendo estatisticamente definido como a quantidade de informação psicométrica que um item contém em todos os pontos ao longo do continuum do traço latente (PASQUALI, 2017).

A FII é uma ferramenta poderosa para análise de itens, permitindo-nos saber não apenas quanta informação um item acumula em um determinado θ , mas também em que θ o item tem a maior quantidade de informações. A FII tem sido o método de análise de item mais comumente usados pelos construtores de teste (BAKER, 2001). A Equação 2.5 define a FII:

$$I(\theta) = a^2 \frac{Q_i(\theta)(P_i(\theta) - c)^2}{P_i(\theta)(1 - c)^2} \quad (2.5)$$

Onde,

a representa o parâmetro de discriminação do item;

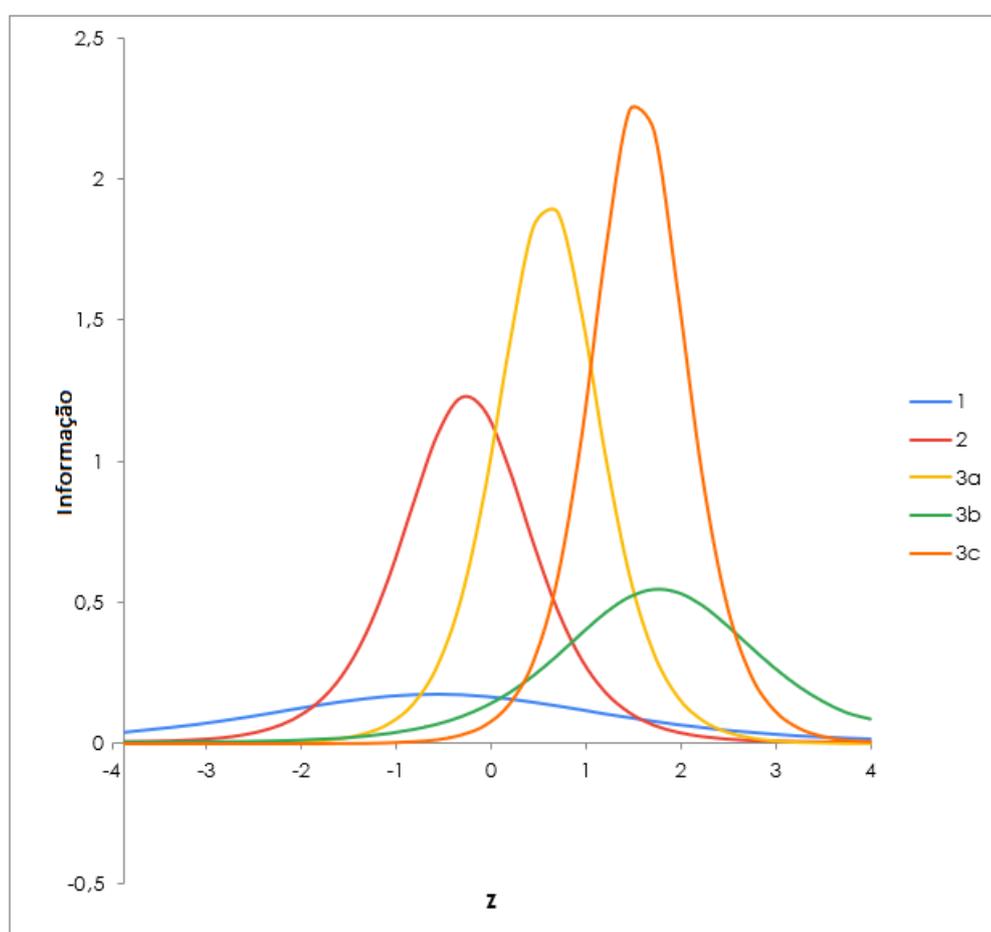
c representa o parâmetro dificuldade do item;

$P_i(\theta)$ é a probabilidade de acerto do item i , segundo o modelo ICC no nível de habilidade θ ;

$Q_i(\theta)$ é a probabilidade de resposta incorreta ao item i , segundo o modelo ICC no nível de habilidade θ .

Usando a fórmula apropriada, a quantidade de informação pode ser calculada para cada nível de habilidade na escala de -4 a +4. A seguir, apresentamos na Figura 2 um exemplo de gráfico que plota algumas FII's.

Figura 2 – FII's exemplo



Fonte: Elaborada pelo autor (2022).

O gráfico da Figura 2 mostra cinco FII's, cada um correspondendo a um item, modelado a partir da Equação 2.5, que usaremos como exemplos para ilustrar o comportamento dos itens. O valor da habilidade Z pode assumir qualquer número real (BAKER, 2001). Geralmente, o eixo x que representa a escala de habilidade Z é classificado de -4 a +4. O eixo y representa o IIF na área onde a habilidade é calculada. As linhas indicam quanta informação cada item fornece em

uma determinada região de traço latente.

No exemplo da Figura 2, o item '3a' fornece menos informação quando a habilidade está em -2, por comparação. No entanto, este item fornece mais informações na área onde a habilidade é média (quando Theta está perto de 0).

3 METODOLOGIA

Neste capítulo, apresentamos o planejamento dos estudos conduzidos para contemplar o objetivo desta pesquisa. Apresentaremos um esboço das etapas para a conclusão desta pesquisa. Ao final desta pesquisa, queremos saber se o instrumento baseado na TRI melhora a capacidade de triagem de novatos em programação para identificar sinais de falhas e evasão antes mesmo deles iniciarem um CS1. Para responder a esta pergunta e concluir este estudo, foram necessárias três etapas:

- **Etapa 1 - Revisão Sistemática de Literatura.** Identificamos as habilidades cognitivas preditoras de programação e as estratégias existentes para medir essas habilidades. Uma revisão sistemática da literatura foi necessária para esta etapa, abrangemos estudos relacionados às habilidades prévias de programação e o levantamento/categorização de estudos que mencionam e incentivam tais habilidades;
- **Etapa 2 - Construção do Banco de itens.** Com base na pesquisa anterior, desenvolvemos itens com base nas teorias psicométricas (discutidas na seção 2.2), levando em consideração os indicadores de habilidades prévias de programação;
- **Etapa 3. Calibração do Banco de Itens.** Em seguida, desenvolvemos instrumentos por meio de agrupamento dos itens desenvolvidos na etapa anterior. Aplicamos e analisamos os instrumentos, verificando a consistência interna e as propriedades psicométricas dos itens (analisando se os instrumentos indicam informações em função do construto específico avaliado).

3.1 Revisão Sistemática de Literatura

Esta seção apresenta o planejamento da Revisão Sistemática da Literatura (RSL) para obter conhecimento sobre as habilidades preditoras relacionadas à programação e como medi-las (KITCHENHAM, 2004). Apresentaremos como respondemos às seguintes perguntas de pesquisa:

- **QP1.** Quais são as habilidades preditoras de programação?
- **QP2.** Como devemos mensurar habilidades preditoras de programação?

Podemos identificar, avaliar e interpretar todas as pesquisas relevantes para uma determinada questão de pesquisa ou área de assunto ou um exemplo de interesse de forma repetitiva ou imparcial por meio de um RSL. RSL difere de uma revisão da literatura na definição do protocolo de avaliação. Os dados obtidos do RSL são pré-requisitos para uma meta-análise quantitativa. Primeiramente, definimos o protocolo de revisão que nos permitiu identificar os objetivos, questões de pesquisa, escopo, estratégia de pesquisa e estudos de seleção de consulta de pesquisa, conforme apresentado a seguir.

3.1.1 *Objetivo e escopo*

Esta RSL tem os seguintes objetivos:

- **Objetivo 1:** identificar as habilidades preditoras envolvidas no ensino de programação para alunos iniciantes;
- **Objetivo 2:** identificar estudos que medem habilidades prévias de programação.

3.1.2 *Estratégia de Pesquisa*

Nossa estratégia de busca consistiu em uma busca online nas 4 (quatro) principais bibliotecas digitais de alta relevância para Engenharia de Software, a saber: IEEE Xplore¹ ACM Digital Library², Science Direct³, e Scopus⁴.

As palavras-chave de busca são essenciais para a qualidade e cobertura dos resultados, por isso devem ser cuidadosamente definidas para busca em bibliotecas digitais online. A consulta é composta por 3 (três) termos, sendo um termo padrão e o outro uma combinação de sinônimos de palavras-chave identificadas. O primeiro termo (A) refere-se a Introdução à Programação e seus sinônimos, o segundo termo (B) refere-se a Programação Iniciante ou CS1, o terceiro termo (C) refere-se a habilidade e seus sinônimos.

Realizamos uma pesquisa piloto no IEEE Xplore para avaliar a consulta. Após algumas tentativas e erros com um intervalo de bancos de dados, selecionamos uma frase de pesquisa combinada que parecia capturar a área de interesse:

((“introduction to programming”OR “programming course”OR “programming language”OR “programming learning”OR “learning programming”OR “programming teaching”OR “teaching programming”) AND (“novice programming”OR CS1) AND (skill OR expertise OR ability OR proficiency OR experience OR art OR technique OR facility OR talent OR intelligence OR craft OR competence OR readiness OR accomplishment OR knack OR ingenuity OR finesse OR aptitude OR dexterity OR cleverness OR quickness OR adroitness OR handedness OR skilfulness))

3.1.3 *Estratégia de Seleção de Estudos*

Definimos explicitamente os critérios de inclusão e exclusão na revisão do protocolo para esta SLR, a saber:

- **Crítérios de inclusão (CI):** estudos que definem habilidades preditoras de programação em CS1.

¹ Ieee xplore library. <https://ieeexplore.ieee.org/>. Accessed: 2020-02-10.

² Acm digital library. <https://dl.acm.org/>. Accessed: 2020-02-10

³ Sciencedirect library. <https://www.sciencedirect.com/>. Accessed: 2020-02-10.

⁴ Scopus library. <https://www.scopus.com/>. Accessed: 2020-02-10

- **Cr terios de exclus o (CE1):** estudos incompletos, indispon veis e/ou duplicados (artigos com vers o igual ou atualizada, mantendo apenas a mais recente);
- **Cr terios de exclus o (CE2):**   um artigo curto ou RSL;
- **Cr terios de exclus o (CE3):** n o est  escrito em ingl s;
- **Cr terios de exclus o (CE4):** publicados antes da  ltima d cada;
- **Cr terios de exclus o (CE5):** estudos que n o definem habilidades de programa o preditiva para alunos iniciantes.

Organizamos nosso processo de sele o de estudos em quatro fases distintas, descritas a seguir:

- **Fase 1:** como sele o preliminar, realizamos as consultas, aplicamos CE4 e CE5 definimos o grupo de estudo que serviu para a segunda fase;
- **Fase 2:** com base nos t tulos, resumos e palavras dos estudos de sele o preliminar, determinamos e mantivemos quais eram os estudos relevantes;
- **Fase 3:** com base nos cr terios de inclus o e exclus o e leitura na  ntegra, revisamos estudos relevantes da fase anterior;
- **Fase 4:** um especialista avaliou e validou os estudos selecionados, com possibilidade de inclus o ou exclus o de estudos.

3.1.4 Extra o dos Dados

Preparamos um formul rio para extrair e sintetizar dados relevantes do estudo para responder  s perguntas de pesquisa do protocolo RSL. A extra o de dados visa resumir dados de estudos prim rios.

O processo de RSL definido na se o anterior resultou em 5.063 artigos encontrados nas quatro bases de dados. Ap s a Fase 1, selecionamos 262 estudos para a Fase 2, na qual identificamos 56 estudos (ALSHAYE; TASIR; JUMAAT, 2019) (ATTALLAH; ILAGURE; CHANG, 2018) (BASNET et al., 2018) (CABO; LANSIQUOT, 2014) (CHAO, 2016) (CHAUDHRY; RASOOL, 2012) (CHETTY; BARLOW-JONES, 2014) (DRACHOVA et al., 2015) (DEKHANE; XU; TSOI, 2019) (DURAK, 2018) (DURAK, 2020) (FIGUEIREDO; GARCIA-PENALVO, 2017) (FIGUEIREDO; GARC A-PE ALVO, 2019) (FIGUEIREDO; NAT LIA; GARC A-PE ALVO, 2016) (GOMES et al., 2017) (HOOSHYAR et al., 2015) (HOOSHYAR; AHMAD; NASIR, 2014) (HOOSHYAR et al., 2014) (HOOSHYAR et al., 2015a) (HOOSHYAR et al., 2015b) (HUEI, 2014) (JAKO ; VERBER, 2017) (JONES; WESTHUIZEN, 2017) (KOULOURI; LAURIA; MACREDIE, 2014) (KUMAR, 2013) (LANSIQUOT; CABO, 2015) (LANSIQUOT;

CABO, 2015) (LISHINSKI et al., 2016) (LÓPEZ et al., 2017) (MALIK, 2016) (MALIK; NEILSON, 2017) (MALIK et al., 2019) (MATHEW; MALIK; TAWAFK, 2019) (MLADENOVÍČ; KRPAŇ; MLADENOVÍČ, 2017) (MULLER; BUTMAN; BUTMAN, 2017) (NIEMELÄ et al., 2017) (PALMER; FLIEGER; HILLENBRAND, 2011) (PAPADOPOULOS; TEGOS, 2012) (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015) (PHILIP; RENUMOL; GOPEEKRISHNAN, 2013) (ROSE, 2016) (SANDS, 2019) (SKALKA; DRLÍK, 2018) (SMETSERS; SMETSERS, 2017) (SOUZA et al., 2019) (SPANGSBERG; FINCHER; DZIALLAS, 2018) (STATTER; ARMONI, 2020) (TAHERI; SASAKI; NGETHA, 2015) (THEVATHAYAN; HAMILTON, 2015) (TOPALLI; CAGILTAY, 2018) (VEERASAMY et al., 2019) (ÇOKLAR; AKCAY, 2018) como relevante após uma leitura cuidadosa do artigo. Nos trabalhos em que o pesquisador principal teve dificuldade em aceitar ou não o artigo nos critérios de inclusão, consultamos professores universitários especializados em ensino de programação para sanar essas dúvidas. Mostramos os detalhes do estudo no Quadro 1 por biblioteca e no Quadro 2 por fase.

Quadro 1 – Detalhes da pesquisa e seleção de estudos por banco de dados

Biblioteca Digital	Resultado de Pesquisa
IEEE Xplore	421
ACM Digital Library	1.807
Science Direct	297
Scopus	2.538
Total estudos	5.063

Fonte: Elaborada pelo autor (2022).

Quadro 2 – Detalhes do estudo da fase de pesquisa

Fase	Descrição	Incluído	Excluído
Fase 1	Procurando resultados	5.063	0
Fase 2	Seleção de título e resumo	262	4.801
Fase 3	Seleção de leitura completa	56	206
Fase 4	Seleção validada por um especialista	56	0

Fonte: Elaborada pelo autor (2022).

3.2 Construção do Banco de Itens

Esta seção apresenta o planejamento da construção de um banco de itens. Apresentaremos como respondemos à seguinte pergunta de pesquisa:

- **QP3.** Os itens que contemplem os indicadores de habilidades preditoras de programação possuem boa análise de conteúdo?

Para responder a esta questão de pesquisa, usamos as teorias psicométricas discutidas na subseção 2.2. Essas teorias possibilitam um conjunto de etapas que devem ser exploradas na construção de itens válidos, conforme especificaremos a seguir.

Junto com a equipe multidisciplinar de especialistas, desenvolvemos um banco de itens para os instrumentos. Desenvolvemos os itens de forma a contemplar os itens fáceis, médios e difíceis de serem resolvidos, portanto divididos em 3 (três) níveis. Projetamos todos os itens de forma que o examinando possa expressar a habilidade respondendo o item. Pensando nisso, iniciamos a redação do item, descrevemos as alternativas (no caso de itens de múltipla escolha) e indicamos a resposta correta.

A elaboração de itens envolve também a definição do tipo e quantidade de itens que irão compor o instrumento. A definição do tipo de item depende da finalidade do instrumento. Portanto, no desenvolvimento dos itens, nossa equipe considerou a quantidade de itens gerenciados pelo instrumento (essas informações dependem exclusivamente do tipo de item). Dessa forma, quanto maior o número de itens no banco, melhor será o instrumento, pois haverá itens mais adequados para um determinado nível de habilidade. No geral, construímos 40 itens para compor a avaliação das habilidades preditivas de programação.

Depois de elaborar os itens, avaliamos teoricamente. Os juízes realizaram análises teóricas, ou seja, por especialistas na área pesquisada. Os juízes verificaram se os itens estão bem compreendidos (análise semântica) e se são adequados para mensurar a habilidade desejada (análise de conteúdo).

A análise semântica verifica se os itens são inteligíveis para todos os assuntos. Os itens precisam ser fáceis de entender para todos, mesmo os assuntos com traços de habilidade mais baixos. A análise semântica visa garantir que a dificuldade de compreensão dos itens não seja um fator complicador que possa interferir na resposta do sujeito.

A análise de conteúdo busca garantir que os itens se referem à habilidade que queremos estimar. O número de juízes pode variar, mas a literatura recomenda que haja pelo menos três (ARAÚJO et al., 2019) (BAKER; KIM, 2017). Dessa forma, utilizamos 3 juízes para analisar o conteúdo das questões. A literatura também indica uma concordância de 80% entre os juízes pode ser uma referência para decidir se o item se refere à habilidade e incluí-la no instrumento. Se houver concordância inferior a 80%, devemos excluir o item do instrumento. No caso de haver três juízes, os três devem concordar em incluir o item no instrumento. Se houver apenas dois juízes, o nível de concordância é de 66,6%, não atingindo os 80% exigidos. Porém, mais importante do que a quantidade é a qualificação do juiz escolhido na área específica da pesquisa.

3.3 Calibração do Banco de Itens

Esta seção apresenta o planejamento da calibração de um banco de itens. Apresentaremos a como responder à seguinte pergunta de pesquisa:

- **QP4.** Os itens que contemplem os indicadores de habilidades preditoras de programação

possuem boas propriedades psicométricas?

- **QP5.** Os instrumentos desenvolvidos a partir do banco de itens apresentam boa confiabilidade?

Para responder a estas questões de pesquisa, usamos as teorias psicométricas discutidas na subseção 2.2. Essas teorias possibilitam um conjunto de etapas que devem ser exploradas na calibração de um banco de itens, conforme especificaremos a seguir.

Após a preparação dos itens, passamos para a etapa de calibração do banco de itens. Nesta etapa, os itens já passaram pela análise semântica e análise de conteúdo dos juízes. Agora, esses itens irão prosseguir para um banco de itens.

Calibrar o banco de itens consiste em aplicar os itens, coletar dados, escolher o modelo de resposta, o método de calibração, projetar e interpretar a escala. Ao aplicar os itens, precisamos ter uma amostra suficiente de respondentes. O tamanho da amostra depende da quantidade de itens do banco, portanto, quanto mais itens, maior é a amostra. No nosso caso, para um banco de dados de 10 itens por habilidade, precisamos de uma amostra de pelo menos 100 respondentes, ou seja, pelo menos dez vezes para cada item. Poderíamos aplicar o formato tradicional em papel e lápis, porém apresentamos em uma versão informatizada para obter a amostra. Todos os assuntos devem responder a todos os itens nesta etapa.

Dessa forma, aplicamos os itens a um grupo de 100 alunos de universidades comunitárias locais na área de computação. Esclarecemos o estudo para professores e administradores. Solicitamos uma amostra de alunos dentro do perfil estabelecido pelo estudo. Foram considerados os seguintes critérios de inclusão para este estudo: Termo de Assentimento (Anexo A) - se o aluno for menor de 18 anos - ou assinatura do Termo de Consentimento Livre e Esclarecido (Anexo B); e, ausência de alterações cognitivas, visuais, psicológicas ou neurológicas. Foram excluídos desses estudos sujeitos que não assinam o termo de autorização e apresentam alterações cognitivas, visuais, psicológicas ou neurológicas.

Consideramos as respostas dos alunos durante a aplicação do instrumento e as transformamos em itens do tipo certo/errado (itens dicotômicos); portanto, na correção, atribuímos 0 ao errado e 1 ao certo. Em seguida, tabulamos os dados e realizamos um estudo para verificar as propriedades psicométricas dos itens. Esse procedimento é uma etapa essencial na construção de qualquer instrumento, pois permite verificar se a escala construída é minimamente adequada para a continuidade do estudo.

A escolha do modelo de resposta consiste em verificar qual modelo da TRI se ajusta ao instrumento. Verificamos se o ajuste foi adequado e, se necessário, substituir o modelo ajustado. Um modelo mal ajustado não fornecerá parâmetros constantes para itens e habilidades. Se as estimativas dos parâmetros dos itens por meio da TRI forem inconsistentes, por exemplo, apresentando valores absurdos ou erro padrão elevado, isso pode ser devido ao tamanho inadequado da amostra.

O método de calibração do banco de itens consiste em estimar os parâmetros do item

usando os critérios TCT e TRI. As análises através dos índices TCT ajudaram a eliminar itens inadequados. Realizamos o processo de calibração dos itens através da TRI e da análise dos parâmetros. Esta análise ajudou na decisão de eliminar itens inadequados. Estimamos os parâmetros usando uma abordagem bayesiana (ver Seção 2.2.2.2). Após eliminar os itens inadequados, refizemos as análises usando o TCT e indicadores de dimensionalidade para verificar se eles não são afetados pela exclusão de itens. Em seguida, executamos novamente o processo de calibração para verificar se os itens restantes são adequados e não são afetados pela exclusão dos outros itens. A TRI considera um banco de itens bem calibrado se as estimativas dos parâmetros dos itens forem adequadas e os erros padrão forem baixos.

Estimativas com parâmetros críticos pelo 3ML implicam que devemos remover o item do banco. Um índice de discriminação abaixo de 0,30 é inadequado para um item ter o poder de diferenciar assuntos com diferentes estimativas de habilidade. Índice de dificuldade abaixo de -2,95 ou acima de 2,95 também são inadequados, pois a escala de habilidade varia de -3 a 3 na prática. Por último, uma probabilidade de acerto pelo acaso acima de 0,40 também é considerada um valor crítico. Em todos esses casos, a literatura recomenda que o item seja excluído do banco de itens para que a estimativa de habilidade não seja comprometida.

Após a calibração final dos itens, devemos fazer alguns cheques bancários. Verificamos se a quantidade de itens no banco de itens foi suficiente para a aplicação do instrumento. De acordo com o objetivo do instrumento, também verificamos se os itens abrangem todo o conteúdo, se estão bem distribuídos e fornecem informações adequadas em todas as extensões do traço latente avaliado (itens fáceis, médios e difíceis). Após calibrar e equalizar os itens, construímos a escala de habilidades. A escala tem como objetivo fornecer uma interpretação qualitativa dos valores obtidos com a aplicação do modelo IRT, permitindo assim a interpretação pedagógica dos valores das competências.

4 ANÁLISE E RESULTADOS

Neste capítulo, apresentamos e discutimos os resultados desta pesquisa que visa melhorar a capacidade de triagem de novatos em programação, fornecendo instrumentos confiáveis para identificar sinais de falhas e evasão de alunos em disciplinas introdutórias de programação.

4.1 Revisão Sistemática de Literatura

Nesta seção apresentamos os principais resultados da Revisão Sistemática da Literatura (RSL) para obter conhecimento sobre as habilidades preditoras relacionadas à programação e como medi-las. Responderemos às seguintes perguntas de pesquisa:

- **QP1.** Quais são as habilidades preditoras de programação?
- **QP2.** Como devemos mensurar habilidades preditoras de programação?

4.1.1 Identificação das Habilidades Preditoras de Programação

Habilidades preditoras são habilidades específicas que precedem um marco ou desenvolvem outras habilidades mais estruturadas. Nesse caso, vários estudos apontam para evidências de que as habilidades que precedem a programação. No Quadro 3, apresentamos estas habilidades e os trabalhos que evidenciam como preditoras. A discussão destas habilidades pode ser vista na Seção 2.1.

Vários estudos apontam evidências de que a habilidade de Resolução de Problemas é uma habilidade de Programação Preditiva (CHAO, 2016) (DURAK, 2020) (HOOSHYAR et al., 2015b) (JAKOŠ; VERBER, 2017) (JONES; WESTHUIZEN, 2017) (KOULOURI; LAURIA; MACREDIE, 2014) (LISHINSKI et al., 2016) (MALIK, 2016) (MALIK; NEILSON, 2017) (MALIK et al., 2019) (MLADENOVÍĆ; KR PAN; MLADENOVÍĆ, 2017) (PHILIP; RENUMOL; GOPEEKRISHNAN, 2013) (SKALKA; DRLÍK, 2018) (TOPALLI; CAGILTAY, 2018) (VEERASAMY et al., 2019). Esses estudos relatam a relação entre as habilidades percebidas pelos alunos na Resolução de Problemas e o desempenho acadêmico em tarefas de Avaliação de Programação Introdutória formativa e somativa.

Além disso, alguns trabalhos visam estudar e melhorar as habilidades de resolução de problemas de estudantes de graduação em Ciência da Computação (CHAUDHRY; RASOOL, 2012) (CHETTY; BARLOW-JONES, 2014) (FIGUEIREDO; GARCIA-PENALVO, 2017) (FIGUEIREDO; GARCÍA-PEÑALVO, 2019) (GOMES et al., 2017) (KUMAR, 2013). Alguns estudos desenvolveram uma estrutura conceitual de aprendizado on-line baseada em problemas para as habilidades de resolução de problemas e programação dos alunos (ALSHAYE; TASSIR; JUMAAT, 2019) (LANSIQUOT; CABO, 2015) (MULLER; BUTMAN; BUTMAN, 2017) (PALMER; FLIEGER; HILLENBRAND, 2011) (PAPADOPOULOS; TEGOS, 2012) (SANDS, 2019).

Quadro 3 – Habilidades preditoras de programação

Habilidade	Estudos
Resolução de Problemas	(ALSHAYE; TASIR; JUMAAT, 2019) (BASNET et al., 2018) (CABO; LANSIQUOT, 2014) (CHAO, 2016) (CHAUDHRY; RASOOL, 2012) (CHETTY; BARLOW-JONES, 2014) (DEKHANE; XU; TSOI, 2019) (DURAK, 2018) (DURAK, 2020) (FIGUEIREDO; GARCIA-PENALVO, 2017) (FIGUEIREDO; GARCÍA-PEÑALVO, 2019) (FIGUEIREDO; NATÁLIA; GARCÍA-PEÑALVO, 2016) (GOMES et al., 2017) (HOOSHYAR et al., 2015) (HOOSHYAR; AHMAD; NASIR, 2014) (HOOSHYAR et al., 2014) (HOOSHYAR et al., 2015a) (HOOSHYAR et al., 2015b) (HUEI, 2014) (JAKOŠ; VERBER, 2017) (JONES; WESTHUIZEN, 2017) (KOULOURI; LAURIA; MACREDIE, 2014) (KUMAR, 2013) (LANSIQUOT; CABO, 2015) (LISHINSKI et al., 2016) (MALIK, 2016) (MALIK; NEILSON, 2017) (MALIK et al., 2019) (MATHEW; MALIK; TAWAFK, 2019) (MLADENOVÍČ; KRPAŇ; MLADENOVÍČ, 2017) (MULLER; BUTMAN; BUTMAN, 2017) (NIEMELÄ et al., 2017) (PALMER; FLIEGER; HILLENBRAND, 2011) (PAPADOPOULOS; TEGOS, 2012) (PHILIP; RENUMOL; GOPEEKRISHNAN, 2013) (ROSE, 2016) (SANDS, 2019) (SKALKA; DRLÍK, 2018) (SMETSERS; SMETSERS, 2017) (SPANGSBERG; FINCHER; DZIALLAS, 2018) (TAHERI; SASAKI; NGETHA, 2015) (THEVATHAYAN; HAMILTON, 2015) (TOPALLI; CAGILTAY, 2018) (VEERASAMY et al., 2019) (ÇOKLAR; AKCAY, 2018)
Pensamento Abstrato	(CHETTY; BARLOW-JONES, 2014) (LÓPEZ et al., 2017) (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015) (PHILIP; RENUMOL; GOPEEKRISHNAN, 2013) (SKALKA; DRLÍK, 2018) (STATTER; ARMONI, 2020)
Raciocínio Matemático	(ATTALLAH; ILAGURE; CHANG, 2018) (DRACHOVA et al., 2015) (GOMES et al., 2017) (SOUZA et al., 2019)
Flexibilidade Cognitiva	(DURAK, 2020)

Fonte: Elaborada pelo autor (2022).

Outros estudos investigaram, por meio de uma pesquisa, a autoeficácia da programação no contexto das habilidades de Resolução de Problemas do ponto de vista de participantes como especialistas, profissionais, estudantes ou professores (ÇOKLAR; AKCAY, 2018) (HUEI, 2014) (MATHEW; MALIK; TAWAFK, 2019) (NIEMELÄ et al., 2017) (THEVATHAYAN; HAMILTON, 2015). E, muitos estudos se concentraram em propor uma solução, abordagem ou instrumento para ajudar os alunos a melhorar as habilidades de resolução de problemas (ALSHAYE; TASIR; JUMAAT, 2019) (BASNET et al., 2018) (CABO; LANSIQUOT, 2014) (DEKHANE; XU; TSOI, 2019) (DURAK, 2018) (FIGUEIREDO; NATÁLIA; GARCÍA-PEÑALVO, 2016) (HOOSHYAR et al., 2014) (HOOSHYAR; AHMAD; NASIR, 2014) (HOOSHYAR et al., 2015) (HOOSHYAR et al., 2015a) (MATHEW; MALIK; TAWAFK, 2019) (MULLER; BUTMAN; BUTMAN, 2017) (PALMER; FLIEGER; HILLENBRAND, 2011) (PAPADOPOULOS; TE-

GOS, 2012) (ROSE, 2016) (SMETSERS; SMETSERS, 2017) (SPANGSBERG; FINCHER; DZIALLAS, 2018) (TAHERI; SASAKI; NGETHA, 2015) (THEVATHAYAN; HAMILTON, 2015).

Alguns estudos criam evidências de que a habilidade de pensamento abstrato é uma habilidade preditiva de programação (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015) (PHILIP; RENUMOL; GOPEEKRISHNAN, 2013) (SKALKA; DRLÍK, 2018) (STATTER; ARMONI, 2020). O pensamento abstrato afeta positivamente a compreensão da parte mais complexa do programa (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015). Além disso, estimular essa habilidade é essencial em Ciência da Computação (PARK; HYUN; HEUILAN, 2015).

Alguns estudos criam evidências de que a habilidade de Raciocínio Matemático é uma habilidade preditora de Programação (GOMES et al., 2017) (SOUZA et al., 2019). Os outros estudos preliminares exploram o impacto da habilidade matemática na aprendizagem de programação no Ensino Superior (ATTALLAH; ILAGURE; CHANG, 2018) e descrevem um processo sistemático de introdução de habilidades de raciocínio matemático no currículo de CS1 e avaliação de quão bem os alunos aprenderam essa habilidade (DRACHOVA et al., 2015).

Recentemente, um estudo (DURAK, 2020) estabeleceu o efeito da flexibilidade cognitiva na programação, indicando que essa habilidade é uma programação preditiva. O estudo mostrou que os alunos que tiveram estimulação das habilidades de flexibilidade cognitiva tiveram melhor desempenho no cronograma de atividades.

4.1.2 Mensuração das Habilidades Preditoras de Programação

Vários estudos usam a TCT como forma de medir as habilidades de programação cognitiva. No entanto, algumas universidades têm adotado outras teorias para essa prática, como a TRI. No Quadro 4, apresentamos os instrumentos que medem as habilidades preditoras de programação. Com base neste levantamento prévio, podemos concluir que não existe um instrumento (mesmo que em papel-e-lápis) que mesure as 4 habilidades preditoras de programação. Surgindo assim, a necessidade de endereçar estudos para este fim.

4.2 Construção do Banco

Nesta seção, apresentamos e discutimos os resultados da etapa de construção do banco de itens. Esta etapa consiste em verificar se os itens apresentam boa análise de conteúdo. Responderemos às seguintes perguntas de pesquisa:

- **QP3.** Os itens que contemplem os indicadores de habilidades preditoras de programação possuem boa análise de conteúdo?

Junto com a equipe multidisciplinar de especialistas, projetamos itens fáceis, médios e difíceis de serem resolvidos. Redigimos os itens, descrevendo as alternativas e indicamos a

Quadro 4 – Instrumentos que medem habilidades preditoras de programação

Habilidade	Instrumentos
Resolução de Problema	Yuan Ze University (CHAO, 2016) Bartin University (DURAK, 2018) (DURAK, 2020) <i>Aladdin and his flying carpet</i> (JAKOŠ; VERBER, 2017) <i>Programmer for International Student Assessment</i> (PISA) (LISHINSKI et al., 2016) Brunel University (KOULOURI; LAURIA; MACREDIE, 2014) Bebras (SPANGSBERG; FINCHER; DZIALLAS, 2018)
Pensamento Abstrato	Jeju National University (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015) Bebras (SPANGSBERG; FINCHER; DZIALLAS, 2018) University of Turku (VEERASAMY et al., 2019)
Raciocínio Matemático	Limestone College (DRACHOVA et al., 2015) University of Coimbra (GOMES et al., 2017) Universidade de São Paulo (SOUZA et al., 2019)
Flexibilidade Cognitiva	Bartin University (DURAK, 2020)

Fonte: Elaborada pelo autor (2022).

resposta correta. Em seguida, analisamos o conteúdo dos itens de forma a garantir que os itens se referem à habilidade que queremos estimar.

Nesta etapa, cinco professores universitários da comunidade local, especialistas em ensino de programação, participaram do estudo. Os professores avaliaram se os itens presentes no instrumento mensuram o constructo do examinado. Para avaliar o grau de concordância entre os juízes, aplicamos a função agree disponível no pacote irr na linguagem R. Esta concordância conseguiu atingir 90%. Mas será que é confiável? Para tanto, a fim de avaliar o grau de confiabilidade entre os juízes, aplicamos o teste estatístico Kappa de Fleiss, com 80% de significância (concordância substancial). Calculamos por meio da função kappam.fleiss presente no pacote irr da linguagem R. Na Tabela 1, apresentamos os resultados do nível de confiabilidade na classificação dos itens entre as habilidades. O nível de confiabilidade entre os juízes é de 0,947 (quase perfeita).

Tabela 1 – Análise de Confiabilidade entre Juízes

Habilidade	Juízes	Itens	Kappa	Z	p
Resolução de Problemas	5	10	0,899	17,971	0,000
Pensamento Abstrato	5	10	0,945	18,904	0,000
Raciocínio Matemático	5	10	1,000	20,000	0,000
Flexibilidade Cognitiva	5	10	0,945	18,904	0,000
Confiabilidade	5	40	0,947	32,800	0.000

Fonte: Elaborada pelo autor (2022).

Desta forma, podemos analisar que os níveis de confiabilidade e concordância entre os juízes é alta. Após a análise das respostas entre os juízes, todos os itens obtiveram índices acima de concordância acima de 80% e foram selecionados para a construção do instrumento das

tarefas de resolução de problemas, pensamento abstrato, raciocínio matemático e flexibilidade cognitiva.

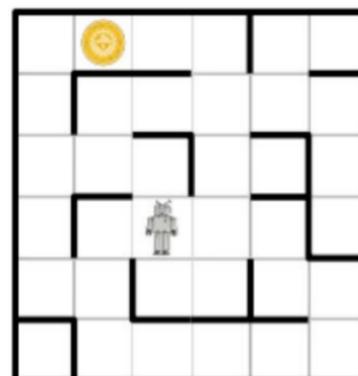
A atividade de resolução de problemas (Fig. 3) permite estimar a habilidade do aluno em conseguir encontrar uma solução para resolver um determinado problema.

Figura 3 – Exemplo de Item da Habilidade Resolução de Problemas

LABIRINTO ESPECIAL

Alguns exploradores do espaço pousaram em um planeta vazio. Da sua nave podiam ver um labirinto com um objeto de prata desconhecido nele. Os exploradores lançaram seu robô no labirinto esperando que ele pudesse dar uma olhada mais de perto no objeto desconhecido. Infelizmente o robô quebrou durante a queda e agora só pode enviar e receber comandos ilegíveis sobre onde ir.

O robô sugere quatro possíveis direções para seguir. Mesmo que as palavras das instruções estejam ilegíveis, ainda há apenas quatro palavras que são diferentes entre si, cada uma indicando norte, oeste, leste ou sul. Ao seguir as instruções, o robô caminhará para um quadrado adjacente (próximo quadrado) conforme as instruções. Que instruções os exploradores devem enviar ao robô para que ele chegue ao objeto de prata?



- Ha' poS poS Ha' Ha' nIH
- Ha' poS poS Ha' nIH Ha'
- Ha' Ha' poS Ha'
- Ha' poS nIH vI'ogh Ha' poS

Fonte: Elaborada pelo autor (2022).

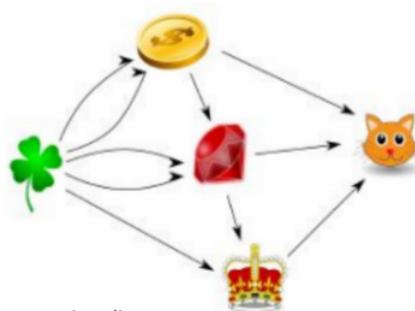
A atividade de pensamento abstrato (Fig. 4) permite estimar a habilidade do aluno em dividir um determinado problema em partes menores e resolvê-los separadamente.

Figura 4 – Exemplo de Item da Habilidade Flexibilidade Cognitiva

ALQUIMISTA

O alquimista pode converter objetos em novos objetos. Ele pode converter:

- Dois trevos em uma moeda;
- Uma moeda e dois trevos em um rubi;
- Um rubi e um trevo em uma coroa;
- Uma moeda, um rubi e uma coroa em um gatinho.



Depois que um objeto foi convertido em um outro objeto, ele desaparece imediatamente.

De quantos trevos o Alquimista precisa para criar um gatinho?

- 5
- 10
- 11
- 12

Fonte: Elaborada pelo autor (2022).

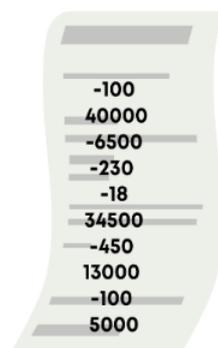
A atividade de raciocínio matemático (Fig. 5) permite estimar a habilidade do aluno na matemática para resolver problemas.

Figura 5 – Exemplo de Item da Habilidade Raciocínio Matemático

DESPESAS DE TRABALHO

Na empresa ABC, o vendedor Robin está vendendo software. Para vender o software, ele organiza reuniões em todo o país. Claro, ele sempre precisa viajar para lá e às vezes até passar a noite. Como Robin está muito ocupado, ele não conseguiu arquivar as despesas do trabalho este mês. Eret, o contador, disse-lhe que poderia consertar para ele, desde que recebesse o preço de todas as despesas junto com os recibos. Robin fez isso, mas entendeu mal: ele forneceu uma lista de todas as despesas e receitas que trouxe no mês passado, com as receitas como inteiros positivos e as despesas como inteiros negativos. Você pode ajudar Eret a descobrir quantas despesas Robin pagou no último mês? Supondo que Robin tenha fornecido a seguinte entrada.

- a) 44.800
- b) 37.402
- c) 36.700
- d) 7.398



Fonte: Elaborada pelo autor (2022).

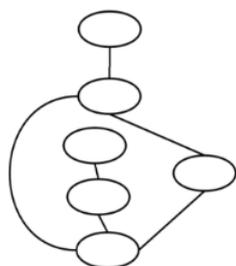
A atividade de flexibilidade cognitiva (Fig. 6) permite estimar a habilidade do aluno em pensar em diferentes soluções para o mesmo problema.

Figura 6 – Exemplo de Item da Habilidade Flexibilidade Cognitiva

TROCA DE PALAVRAS

Na sua lição de casa, Tomás teve que escrever palavras em inglês nos cartões e conectá-las com elásticos. O professor de inglês pediu para ligar quaisquer duas palavras que se diferenciam em exatamente uma letra. Tomás fez isso, como você pode ver na imagem à direita.

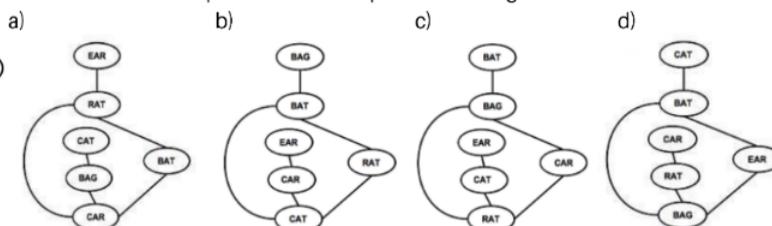
Tomás parou de estudar por um tempo, e quando voltou, ele teve uma surpresa. Pedro, seu irmão mais novo, tinha apagado todas as palavras! Além disso, os cartões foram completamente misturados, como você pode ver na imagem à esquerda.



É importante ressaltar que as linhas com elástico ainda estavam conectadas como antes.

Tomás tem certeza que ele poderia colocar as palavras de volta no lugar correto.

Questão: Qual das respostas contém as palavras nos lugares corretos?



Fonte: Elaborada pelo autor (2022).

4.3 Calibração do Banco de Itens

Nesta seção, apresentamos e discutimos os resultados da etapa de calibração dos itens. Esta etapa consiste em verificar as propriedades psicométricas dos itens do instrumento e analisar se a escala construída se encontra minimamente ajustada para dar continuidade ao estudo. Responderemos às seguintes perguntas de pesquisa:

- **QP4.** Os itens que contemplem os indicadores de habilidades preditoras de programação possuem boas propriedades psicométricas?
- **QP5.** Os instrumentos desenvolvidos a partir do banco de itens apresentam boa confiabilidade?

Apresentamos os itens aos participantes com objetivo de obter informações como acerto/erro do item e o tempo de resposta nas atividades. Para este estudo, participaram 100 estudantes oriundos de universidades e institutos locais de Campina Grande, Paraíba, Brasil. Aplicamos o instrumento de forma virtual, os alunos tinham um prazo de 24 horas para responder os itens, salientando que a resposta deveria ser feita sem consulta online ou com algum monitor da disciplina. Consideramos as respostas dos alunos durante a aplicação, e as transformamos em itens do tipo certo/errado (itens dicotômicos), portanto, na correção foi atribuído 0 ao errar e 1 ao acertar.

Os dados coletados nesta fase foram analisados pela TRI com auxílio da ferramenta Excel disponível em: <http://psychometricon.net/libirt/> para análise e ajuste do ML3 pela estimativa marginal de máxima verossimilhança, com finalidade de verificar: i) a consistência interna do instrumento (apresentada na Tabela 2, a partir dos dados obtidos pelo alfa de Cronbach, conclui-se que o instrumento é confiável para todas as atividades apresentadas) e; ii) a estimativa dos parâmetros dos itens das tarefas do instrumento que será apresentada a seguir.

Tabela 2 – Consistência Interna do Instrumento

Habilidade	Sujeitos	Itens	Pontuação Média	Desvio Padrão	Cronbach Alpha
Resolução de Problema	100	10	7,174	2,648	0,854
Pensamento Abstrato	100	10	6,652	2,547	0,786
Raciocínio Matemático	100	10	6,522	2,534	0,757
Flexibilidade Cognitiva	100	10	6,261	2,594	0,791

Fonte: Elaborada pelo autor (2022).

Interpretamos a distribuição das respostas dos alunos em cada item das tarefas por meio do 3ML, além disso consideramos a proporção de acertos e a correlação ponto bisserial entre a resposta correta no item e a pontuação total na tarefa.

4.3.1 Resolução de Problemas

Na Tabela 3, apresentamos os itens que compõem a habilidade de Resolução de Problemas com os respectivos parâmetros. E na Tabela 4, os respectivos índices de ajustes ao 3ML.

Tabela 3 – Habilidade Resolução de Problemas - Todos os Itens Calibrados

Id	a	b	c	Proporção de Acertos	Correlação Ponto Bis-serial
Res-001	3,719	-0,885	0,143	0,783	0,771
Res-002	3,367	-0,745	0,169	0,783	0,670
Res-003	1,904	-0,367	0,145	0,652	0,637
Res-004	0,815	-1,926	0,165	0,826	0,340
Res-005	2,014	0,633	0,127	0,391	0,486
Res-006	2,015	0,996	0,123	0,304	0,426
Res-007	1,478	-0,817	0,158	0,739	0,522
Res-008	2,568	-1,805	0,160	0,913	0,595
Res-009	2,568	-1,805	0,160	0,913	0,595
Res-010	2,689	-1,371	0,165	0,870	0,694

Fonte: Elaborada pelo autor (2022).

Tabela 4 – Habilidade Resolução de Problemas - Todos os Itens Ajustados

Id	Chi-square	Grau de liberdade	P-value
Res-001	1,122	10	1,000
Res-002	0,726	10	1,000
Res-003	0,920	10	1,000
Res-004	0,548	10	1,000
Res-005	1,410	10	0,999
Res-006	1,368	10	0,999
Res-007	0,301	10	1,000
Res-008	0,541	10	1,000
Res-009	0,541	10	1,000
Res-010	0,721	10	1,000

Fonte: Elaborada pelo autor (2022).

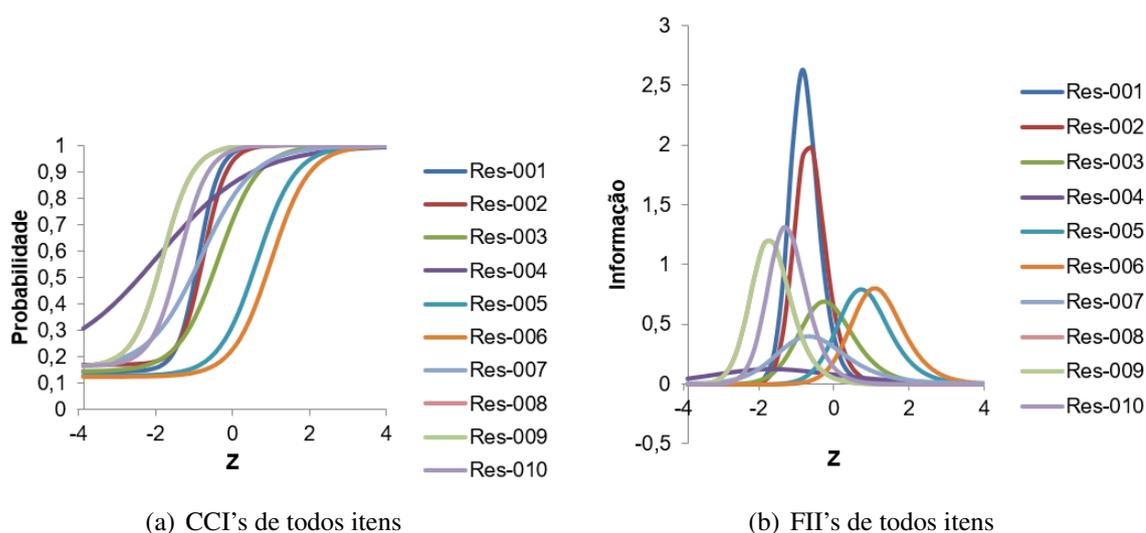
Após a constatação do modelo logístico 3ML, não encontramos valores críticos para os parâmetros estimados, todos os itens da habilidade Resolução de Problemas possuem valores superiores a 0,30 para o índice de discriminação; para o índice de dificuldade valores entre 3,95

e -3,95; e, a probabilidade de chute abaixo de 0,40. Além disso, os resultados revelaram que esta tarefa apresenta itens fáceis (com índices acima de 75%), itens moderados (com índices entre 50 a 75%) e itens difíceis (com índices abaixo de 50%).

As correlações ponto-biserial revelaram tendência de escolha da opção errada pelos participantes que obtiveram escores mais altos no teste para os itens Res-004, Res-005 e Res-006. Apesar disso, todos os itens se ajustaram adequadamente ao 3ML, logo apresentam boa confiabilidade e bom índice de separação da habilidade.

Apresentamos na Figura 7(a) a representação gráfica das CCI's dos itens para a habilidade de Resolução de Problemas, em que se destacam os valores extremos dos parâmetros discriminação, dificuldade e o acerto ao acaso. E na Figura 7(b), apresentamos a representação gráfica das FII's dos itens para a habilidade de Resolução de Problemas, em que se destaca o quanto de informação cada item fornece em uma determinada região do traço latente.

Figura 7 – CCI's e FII's dos itens da Habilidade Resolução de Problemas.



Fonte: Elaborada pelo autor (2022).

Podemos observar pelas CCI's que o item Res-004 é o menos discriminativo e o mais fácil de acertar. Já o item Res-001 é o mais discriminativo. O mais difícil de acertar é o item Res-006, e por curiosidade, é também o item com menores chances de acertar pelo acaso. Já o item Res-002 é o item mais fácil de acertar pelo acaso. Com relação à probabilidade de acerto pelo acaso, esta habilidade apresenta índices abaixo de 16%, bem abaixo do esperado, uma vez que os itens desta habilidade possuem quatro alternativas cada, podendo-se afirmar matematicamente que a probabilidade de um estudante de baixa habilidade acertar o item é aproximadamente igual 25%.

Em relação à Função de Informação, o item Res-002 oferece mais informação para avaliar sujeitos de habilidade mediana, ou seja, em um instrumento informatizado, esse item seria o primeiro a ser escolhido, se o instrumento situar a habilidade Theta em 0 como a habilidade média.

4.3.2 *Pensamento Abstrato*

Na Tabela 5, apresentamos os itens que compõem a habilidade de Pensamento Abstrato com os respectivos parâmetros. E na Tabela 6, os respectivos índices de ajustes ao 3ML.

Tabela 5 – Habilidade Pensamento Abstrato - Todos os Itens Calibrados

Id	a	b	c	Proporção de Acertos	Correlação Ponto Bis-serial
Abs-001	1,712	-0,191	0,145	0,609	0,573
Abs-002	2,182	-2,375	0,164	0,957	0,498
Abs-003	1,701	-0,170	0,149	0,609	0,529
Abs-004	3,362	-1,332	0,152	0,870	0,692
Abs-005	2,461	0,127	0,135	0,522	0,607
Abs-006	0,601	-0,153	0,172	0,609	0,124
Abs-007	2,965	-1,707	0,158	0,913	0,619
Abs-008	1,194	-0,425	0,154	0,652	0,393
Abs-009	1,378	1,144	0,151	0,348	0,333
Abs-010	1,611	-0,010	0,144	0,565	0,499

Fonte: Elaborada pelo autor (2022).

Tabela 6 – Habilidade Pensamento Abstrato - Todos os Itens Ajustados

Id	Chi-square	Grau de liberdade	P-value
Abs-001	1,008	10	1,000
Abs-002	0,628	8	1,000
Abs-003	0,765	10	1,000
Abs-004	1,001	10	1,000
Abs-005	0,947	10	1,000
Abs-006	0,794	10	1,000
Abs-007	0,849	9	1,000
Abs-008	1,023	10	1,000
Abs-009	0,725	10	1,000
Abs-010	1,002	10	1,000

Fonte: Elaborada pelo autor (2022).

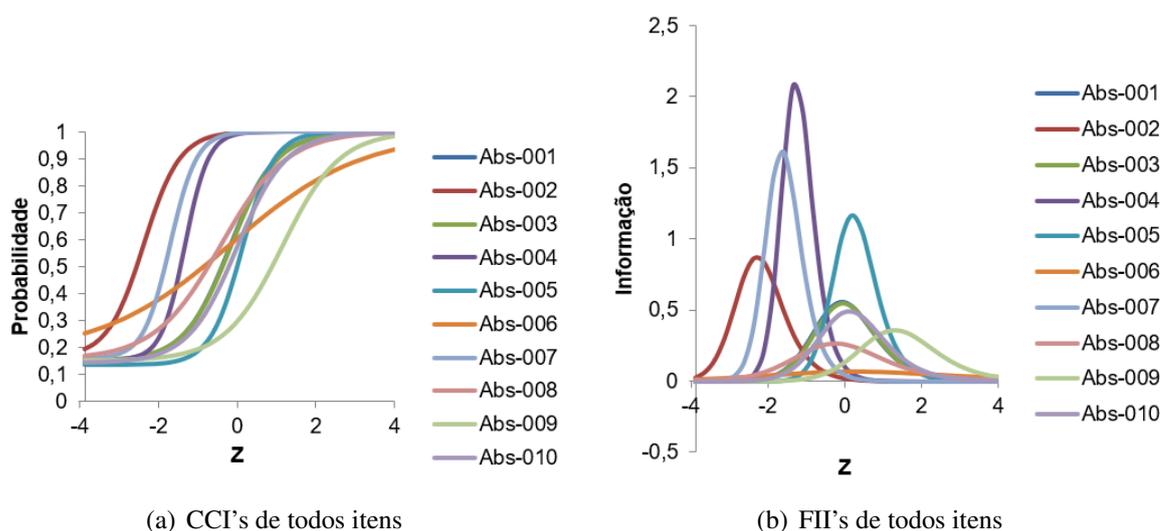
Após a constatação do modelo logístico 3ML, não encontramos valores críticos para os parâmetros estimados, todos os itens da habilidade Pensamento Abstrato possuem valores superiores a 0,30 para o índice de discriminação; para o índice de dificuldade valores entre 3,95 e -3,95; e, a probabilidade de chute abaixo de 0,40. Além disso, os resultados revelaram que está

tarefa apresenta itens fáceis (com índices acima de 75%), itens moderados (com índices entre 50 a 75%) e itens difíceis (com índices abaixo de 50%).

As correlações ponto-bisserial revelaram tendência de escolha da opção errada pelos participantes que obtiveram escores mais altos no teste para os itens Abs-006, Abs-008, Abs-009. Apesar disso, todos os itens se ajustaram adequadamente ao 3ML, logo apresentam boa confiabilidade e bom índice de separação da habilidade.

Apresentamos na Figura 8(a) a representação gráfica das CCI's dos itens para a habilidade de Pensamento Abstrato, em que se destacam os valores extremos dos parâmetros discriminação, dificuldade e o acerto ao acaso. E na Figura 8(b), apresentamos a representação gráfica das FII's dos itens para a habilidade de Pensamento Abstrato, em que se destaca o quanto de informação cada item fornece em uma determinada região do traço latente.

Figura 8 – CCI's e FII's dos itens da Habilidade Pensamento Abstrato.



Fonte: Elaborada pelo autor (2022).

Podemos observar pelas CCI's que o item Abs-006 é o menos discriminativo e o mais fácil de acertar pelo chute. O item Abs-004 é o mais discriminativo. Já o item mais fácil de acertar é o Abs-002 e o mais difícil é o Abs-009. Levando em consideração à probabilidade de acerto pelo acaso, o item mais difícil de ser "chutado" é o item Abs-005. Esta tarefa apresenta índices de "chute" abaixo de 17%, bem abaixo do esperado, uma vez que os itens desta habilidade possui quatro alternativas cada, podendo-se afirmar matematicamente que a probabilidade de um estudante de baixa habilidade acertar o item é aproximadamente igual 25%.

Em relação à Função de Informação do Item, o item Abs-005 oferece mais informação para avaliar sujeitos de habilidade mediana, ou seja, e um instrumento informatizado, esse item seria o primeiro a ser escolhido, se o instrumento situar a habilidade Theta em 0 como a habilidade média.

4.3.3 Raciocínio Matemático

Na Tabela 7, apresentamos os itens que compõem a habilidade de Raciocínio Matemático com os respectivos parâmetros. E na Tabela 8, os respectivos índices de ajustes ao 3ML.

Tabela 7 – Habilidade Raciocínio Matemático - Todos os Itens Calibrados

Id	a	b	c	Proporção de Acertos	Correlação Ponto Bis-serial
Mat-001	2,622	-1,092	0,159	0,826	0,535
Mat-002	1,445	-0,001	0,151	0,565	0,489
Mat-003	0,676	0,167	0,169	0,565	0,205
Mat-004	1,203	0,956	0,152	0,391	0,297
Mat-005	1,675	-0,997	0,161	0,783	0,539
Mat-006	2,716	-1,078	0,160	0,826	0,645
Mat-007	0,974	-0,965	0,170	0,739	0,323
Mat-008	1,684	-0,478	0,172	0,696	0,399
Mat-009	3,887	-0,531	0,138	0,696	0,678
Mat-010	0,834	1,080	0,169	0,435	0,226

Fonte: Elaborada pelo autor (2022).

Tabela 8 – Habilidade Raciocínio Matemático - Todos os Itens Ajustados

Id	Chi-square	Grau de liberdade	P-value
Mat-001	0,397	9	1,000
Mat-002	0,676	9	1,000
Mat-003	0,583	9	1,000
Mat-004	0,264	9	1,000
Mat-005	0,413	9	1,000
Mat-006	0,438	9	1,000
Mat-007	0,892	9	1,000
Mat-008	0,531	9	1,000
Mat-009	1,049	9	0,999
Mat-010	0,430	9	1,000

Fonte: Elaborada pelo autor (2022).

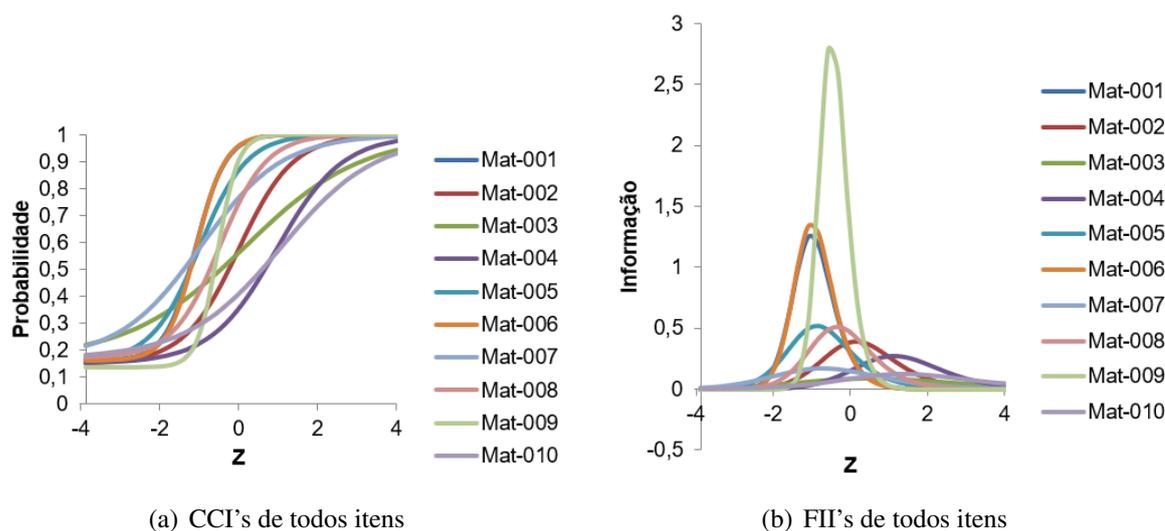
Após a constatação do modelo logístico 3ML, não encontramos valores críticos para os parâmetros estimados, todos os itens da habilidade Raciocínio Matemático possuem valores superiores a 0,30 para o índice de discriminação; para o índice de dificuldade valores entre 3,95 e -3,95; e, a probabilidade de chute abaixo de 0,40. Além disso, os resultados revelaram que está

tarefa apresenta itens fáceis (com índices acima de 75%), itens moderados (com índices entre 50 a 75%) e itens difíceis (com índices abaixo de 50%).

As correlações ponto-bisserial revelaram tendência de escolha da opção errada pelos participantes que obtiveram escores mais altos no teste para os itens Mat-003, Mat-004 e Mat-010. Apesar disso, todos os itens se ajustaram adequadamente ao 3ML, logo apresentam boa confiabilidade e bom índice de separação da habilidade.

Apresentamos na Figura 9(a) a representação gráfica das CCI's dos itens para a habilidade de Raciocínio Matemático, em que se destacam os valores extremos dos parâmetros discriminação, dificuldade e o acerto ao acaso. E na Figura 9(b), apresentamos a representação gráfica das FII's dos itens para a habilidade de Raciocínio Matemático, em que se destaca o quanto de informação cada item fornece em uma determinada região do traço latente.

Figura 9 – CCI's e FII's dos itens da Habilidade Raciocínio Matemático.



Fonte: Elaborada pelo autor (2022).

Podemos observar pelas CCI's que o item Mat-003 é o menos discriminativo. Já o item mais discriminativo é o Mat-009, é este também o que menor chances de acertar pelo chute. O item mais fácil de acertar é o Mat-001 e o mais difícil é o Mat-010. Levando em consideração à probabilidade de acerto pelo acaso, o item mais fácil de ser "chutado" é o item Mat-008. Esta tarefa apresenta índices de "chute" abaixo de 17%, bem abaixo do esperado, uma vez que os itens desta habilidade possui quatro alternativas cada, podendo-se afirmar matematicamente que a probabilidade de um estudante de baixa habilidade acertar o item é aproximadamente igual 25%.

Em relação à Função de Informação do Item, o item Mat-003 oferece mais informação para avaliar sujeitos de habilidade mediana, ou seja, e um instrumento informatizado, esse item seria o primeiro a ser escolhido, se o instrumento situar a habilidade Theta em 0 como a habilidade média.

4.3.4 Flexibilidade Cognitiva

Na Tabela 9, apresentamos os itens que compõem a habilidade de Flexibilidade Cognitiva com os respectivos parâmetros. E na Tabela 10, os respectivos índices de ajustes ao 3ML.

Tabela 9 – Habilidade Flexibilidade Cognitiva - Todos os Itens Calibrados

Id	a	b	c	Proporção de Acertos	Correlação Ponto Bis-serial
Flex-001	2,004	-1,810	0,164	0,913	0,486
Flex-002	0,599	1,400	0,169	0,435	0,169
Flex-003	2,032	0,123	0,141	0,522	0,624
Flex-004	1,658	-0,771	0,160	0,739	0,468
Flex-005	2,436	-1,715	0,162	0,913	0,553
Flex-006	1,804	-0,703	0,168	0,739	0,468
Flex-007	2,773	-0,058	0,140	0,565	0,577
Flex-008	2,744	0,647	0,115	0,348	0,588
Flex-009	0,817	-1,143	0,164	0,739	0,291
Flex-010	2,553	0,757	0,130	0,348	0,543

Fonte: Elaborada pelo autor (2022).

Tabela 10 – Habilidade Flexibilidade Cognitiva - Todos os Itens Ajustados

Id	Chi-square	Grau de liberdade	P-value
Flex-001	0,308	10	1,000
Flex-002	1,321	10	0,999
Flex-003	0,858	10	1,000
Flex-004	0,232	10	1,000
Flex-005	0,431	9	1,000
Flex-006	0,507	10	1,000
Flex-007	0,765	10	1,000
Flex-008	1,606	10	0,999
Flex-009	0,954	10	1,000
Flex-010	0,726	10	1,000

Fonte: Elaborada pelo autor (2022).

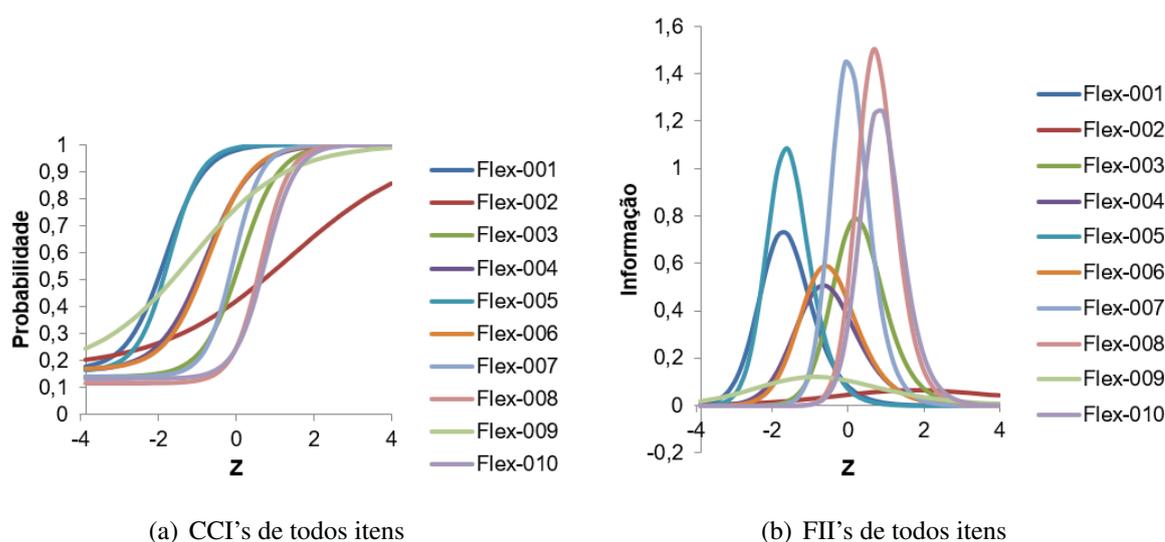
Após a constatação do modelo logístico 3ML, não encontramos valores críticos para os parâmetros estimados, todos os itens da habilidade Flexibilidade Cognitiva possuem valores superiores a 0,30 para o índice de discriminação; para o índice de dificuldade valores entre 3,95

e -3,95; e, a probabilidade de chute abaixo de 0,40. Além disso, os resultados revelaram que esta tarefa apresenta itens fáceis (com índices acima de 75%), itens moderados (com índices entre 50 a 75%) e itens difíceis (com índices abaixo de 50%).

As correlações ponto-biserial revelaram tendência de escolha da opção errada pelos participantes que obtiveram escores mais altos no teste para os itens Flex-002 e Flex-009. Apesar disso, todos os itens se ajustaram adequadamente ao 3ML, logo apresentam boa confiabilidade e bom índice de separação da habilidade.

Apresentamos na Figura 10(a) a representação gráfica das CCI's dos itens para a habilidade de Flexibilidade Cognitiva, em que se destacam os valores extremos dos parâmetros discriminação, dificuldade e o acerto ao acaso. E na Figura 10(b), apresentamos a representação gráfica das FII's dos itens para a habilidade de Flexibilidade Cognitiva, em que se destaca o quanto de informação cada item fornece em uma determinada região do traço latente.

Figura 10 – CCI's e FII's dos itens da Habilidade Flexibilidade Cognitiva.



Fonte: Elaborada pelo autor (2022).

Podemos observar pelas CCI's que o item Flex-002 é o menos discriminativo, o mais difícil de acertar e o que mais chances de um examinado acertar pelo chute. O item Flex-007 é o mais discriminativo. O item mais fácil é o Flex-001. E, o que tem menos probabilidade de acertar pelo chute é o Flex-008. Esta tarefa apresenta índices de "chute" abaixo de 17%, bem abaixo do esperado, uma vez que os itens desta habilidade possui quatro alternativas cada, podendo-se afirmar matematicamente que a probabilidade de um estudante de baixa habilidade acertar o item é aproximadamente igual 25%.

Em relação à Função de Informação do Item, o item Flex-007 oferece mais informação para avaliar sujeitos de habilidade mediana, ou seja, e um instrumento informatizado, esse item seria o primeiro a ser escolhido, se o instrumento situar a habilidade Theta em 0 como a habilidade média.

5 CONSIDERAÇÕES FINAIS E SUGESTÕES PARA TRABALHOS FUTUROS

Este trabalho partiu da necessidade de fornecer uma melhor compreensão sobre as habilidades prévias à programação e destacar estratégias para mensurar tais habilidades, permitindo que os professores possam diagnosticar/rastrear precocemente falhas de novatos em CS1. Para tanto, contribuímos para melhorar a capacidade de triagem de novatos em programação, fornecendo instrumentos confiáveis para identificar sinais de falhas e evasão de alunos em CS1.

A principal questão de pesquisa deste estudo é: o instrumento baseado na Teoria de Mensuração melhora a capacidade de triagem de novatos em programação para identificar sinais de falhas e evasão antes mesmo deles iniciarem um CS1? A fim de entender melhor se é possível identificar se um novato em programação tende a falhar/evadir ou não de um CS1, esta pesquisa obteve os seguintes resultados em cada questão específica dentro do escopo mais amplo:

- **(QP1.)** As habilidades preditoras de programação são Resolução de Problemas, Pensamento Abstrato, Raciocínio Matemático e Flexibilidade Cognitiva;
- **(QP2.)** As habilidades de programação podem ser mensuradas por meio das Teorias de Mensuração: Teoria Clássica dos Testes e Teoria de Resposta ao Item;
- **(QP3.)** Desenvolvemos itens com boa análise de conteúdo que contemplem as habilidades preditoras de programação;
- **(QP4.)** Os itens que contemplem os indicadores de habilidades preditoras de programação possuem boas propriedades psicométricas;
- **(QP5.)** Os instrumentos desenvolvidos a partir do banco de itens apresentam boa confiabilidade em avaliações profissionais.

Mediante dos resultados, pretende-se realizar novos estudos. Entre as diversas possibilidades, destacam-se:

- Analisar as habilidades Resolução de Problemas, Pensamento Abstrato, Raciocínio Matemático e Flexibilidade Cognitiva em triagens feitas previamente ao CS1 em universidades locais e regionais;
- Normatizar o instrumento a fim de viabilizar uma interpretação dos dados (análise estatística) com objetivo de fornecer um instrumento normatizado;
- Verificar o impacto do instrumento em relação a diminuição das taxas de evasão e reprovação em CS1.

REFERÊNCIAS

- ALSHAYE, I.; TASIR, Z.; JUMAAT, N. F. The conceptual framework of online problem-based learning towards problem-solving ability and programming skills. In: IEEE, PULAU PINANG, MALAYSIA. *Proceedings of the Conference on e-Learning, e-Management e-Services (IC3e)*. [S.l.], 2019. Citado 5 vezes nas páginas 14, 18, 30, 35 e 36.
- ANDRADE, D. F.; TAVARES, H. R.; VALLE, R. C. *Teoria da Resposta ao Item: Conceitos e Aplicações*. [S.l.]: ABE, São Paulo, 2000. Citado na página 23.
- ANDRADE, J. M.; LAROS, J. A.; GOUVEIA, V. V. O uso da teoria de resposta ao item em avaliações educacionais: Diretrizes para pesquisadores. *Avaliação Psicológica*, Instituto Brasileiro de Avaliação Psicológica, v. 9, n. 3, 2010. Citado na página 22.
- ARAÚJO, A. L. S. O. et al. Metodologia de pesquisa em informática na educação: Abordagem quantitativa de pesquisa. In: _____. Porto Alegre: SBC, 2019. cap. Teoria de Resposta ao Item. Citado 6 vezes nas páginas 20, 21, 22, 23, 24 e 32.
- ATTALLAH, B.; ILAGURE, Z.; CHANG, Y. K. The impact of competencies in mathematics and beyond on learning computer programming in higher education. In: IEEE, DUBAI, UNITED ARAB EMIRATES. *Proceedings of the Information Technology Trends (ITT)*. [S.l.], 2018. Citado 4 vezes nas páginas 20, 30, 36 e 37.
- BAKER, F. B. *The Basics of Item Response Theory*. [S.l.]: ERIC, 2001. Citado 4 vezes nas páginas 21, 23, 25 e 26.
- BAKER, F. B.; KIM, S.-H. *The basics of item response theory using R*. [S.l.]: Springer, 2017. Citado na página 32.
- BASNET, R. B. et al. Exploring computer science students' continuance intentions to use kattis. *Education and Information Technologies*, Springer, v. 23, n. 3, 2018. Citado 2 vezes nas páginas 30 e 36.
- BORGES, M. A. Avaliação de uma metodologia alternativa para a aprendizagem de programação. In: *VIII Workshop de Educação em Computação-WEI*. [S.l.: s.n.], 2000. p. 15. Citado na página 14.
- CABO, C.; LANSIQUOT, R. D. Synergies between writing stories and writing programs in problem-solving courses. In: IEEE, MADRID, SPAIN. *Proceedings of the Frontiers in Education Conference (FIE)*. [S.l.], 2014. Citado 3 vezes nas páginas 15, 30 e 36.
- CASPERSEN, M. E.; CHRISTENSEN, H. B. Here, there and everywhere- on the recurring use of turtle graphics in cs1. In: *Proceedings of the ACM International Conference Proceeding Series*. [S.l.: s.n.], 2000. v. 8. Citado na página 15.
- CHAO, P. Y. Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. *Computers Education*, Elsevier, v. 95, 2016. Citado 4 vezes nas páginas 30, 35, 36 e 38.
- CHAUDHRY, N.; RASOOL, G. A case study on improving problem solving skills of undergraduate computer science students. *World Applied Sciences Journal*, Springer, Pakistan, v. 20, n. 1, 2012. Citado 4 vezes nas páginas 18, 30, 35 e 36.

CHETTY, J.; BARLOW-JONES, G. Novice students and computer programming: Toward constructivist pedagogy. *Mediterranean Journal of Social Sciences*, MCSER Publishing, Rome, Italy, v. 5, n. 14, 2014. Citado 3 vezes nas páginas 30, 35 e 36.

DEKHANE, S.; XU, X.; TSOI, M. Y. Mobile app development to increase student engagement and problem solving skills. *Journal of Information Systems Education*, Information Systems and Computing Academic Professionals, Wilmington, NC, United States, v. 24, n. 4, 2019. Citado 2 vezes nas páginas 30 e 36.

DRACHOVA, S. V. et al. Teaching mathematical reasoning principles for software correctness and its assessment. *ACM Transactions on Computing Education (TOCE)*, ACM New York, NY, United States, v. 15, n. 3, 2015. Citado 5 vezes nas páginas 20, 30, 36, 37 e 38.

DURAK, H. Y. The effects of using different tools in programming teaching of secondary school students on engagement, computational thinking and reflective thinking skills for problem solving. *Technology, Knowledge and Learning*, Springer, 2018. Citado 4 vezes nas páginas 20, 30, 36 e 38.

DURAK, H. Y. Modeling different variables in learning basic concepts of programming in flipped classrooms. *Journal of Educational Computing Research*, v. 58, n. 1, 2020. Citado 6 vezes nas páginas 20, 30, 35, 36, 37 e 38.

FIGUEIREDO, J.; GARCIA-PENALVO, F. J. Improving computational thinking using follow and give instructions. In: ACM CÁDIZ, SPAIN. *Proceedings of the Technological Ecosystems for Enhancing Multiculturality (TEEM)*. [S.l.], 2017. Citado 3 vezes nas páginas 30, 35 e 36.

FIGUEIREDO, J.; GARCÍA-PEÑALVO, F. J. Teaching and learning strategies of programming for university courses. In: ACM LEÓN, SPAIN. *Proceedings of the Technological Ecosystems for Enhancing Multiculturality (TEEM)*. [S.l.], 2019. Citado 3 vezes nas páginas 30, 35 e 36.

FIGUEIREDO, J.; NATÁLIA, G.; GARCÍA-PEÑALVO, F. J. Ne-course for learning programming. In: ACM SALAMANCA, SPAIN. *Proceedings of the Technological Ecosystems for Enhancing Multiculturality (TEEM)*. [S.l.], 2016. Citado 2 vezes nas páginas 30 e 36.

FOX, J. P.; GLAS, C. A. W. Bayesian estimation of a multilevel irt model using gibbs sampling. *Psychometrika*, Springer, v. 66, n. 2, 2001. Citado na página 25.

GOMES, A. et al. Student's characteristics and programming learning—a macanese perspective. In: IEEE, HONG KONG, CHINA. *Proceedings of the International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*. [S.l.], 2017. Citado 5 vezes nas páginas 30, 35, 36, 37 e 38.

HOOSHYAR, D. et al. Flowchart-based bayesian intelligent tutoring system for computer programming. In: IEEE, KUALA LUMPUR, MALAYSIA. *Proceedings of the International Conference on Smart Sensors and Application (ICSSA)*. [S.l.], 2015. Citado 3 vezes nas páginas 18, 30 e 36.

HOOSHYAR, D.; AHMAD, R. B.; NASIR, M. H. N. M. A framework for automatic text-to-flowchart conversion: A novel teaching aid for novice programmers. In: IEEE, BANDUNG, INDONESIA. *Proceedings of the International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*. [S.l.], 2014. Citado 2 vezes nas páginas 30 e 36.

HOOSHYAR, D. et al. Flowchart-based approach to aid novice programmers: A novel framework. In: IEEE, KUALA LUMPUR, MALAYSIA. *Proceedings of the International Conference on Computer and Information Sciences (ICCOINS)*. [S.l.], 2014. Citado 2 vezes nas páginas 30 e 36.

HOOSHYAR, D. et al. Flowchart-based programming environments for improving comprehension and problem-solving skill of novice programmers: A survey. *International Journal of Advanced Intelligence Paradigms*, v. 7, n. 1, 2015. Citado 2 vezes nas páginas 30 e 36.

HOOSHYAR, D. et al. A flowchart-based multi-agent system for assisting novice programmers with problem solving activities. *Malaysian Journal Of Computer Science*, v. 28, n. 2, 2015. Citado 3 vezes nas páginas 30, 35 e 36.

HUEI, Y. C. Benefits and introduction to python programming for freshmen students using inexpensive robots. In: IEEE, WELLINGTON, NEW ZEALAND. *Proceedings of the International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*. [S.l.], 2014. Citado 2 vezes nas páginas 30 e 36.

JAKOŠ, F.; VERBER, D. Learning basic programming skills with educational games: A case of primary schools in slovenia. *Journal of Educational Computing Research*, v. 55, n. 5, 2017. Citado 4 vezes nas páginas 30, 35, 36 e 38.

JONES, G. B.; WESTHUIZEN, D. V. Pre-entry attributes are thought to influence the performance of students in computer programming. In: SPRINGER, CHAM. *Proceedings of the Southern African Computer Lecturers' Association*. [S.l.], 2017. Citado 3 vezes nas páginas 30, 35 e 36.

KITCHENHAM, B. Procedures for performing systematic reviews. *Keele, UK, Keele University*, v. 33, n. 2004, p. 1–26, 2004. Citado na página 28.

KOLEN, M. J.; TONG, Y. Psychometric properties of irt proficiency estimates. *Educational Measurement: Issues and Practice*, Wiley Online Library, v. 29, n. 2, 2010. Citado na página 25.

KOULOURI, T.; LAURIA, S.; MACREDIE, R. D. Teaching introductory programming: A quantitative evaluation of different approaches. *ACM Transactions on Computing Education (TOCE)*, ACM New York, NY, United States, v. 14, n. 4, 2014. Citado 5 vezes nas páginas 19, 30, 35, 36 e 38.

KUMAR, A. N. A mid-career review of teaching computer science i. In: ACM DENVER, COLORADO, UNITED STATES. *Proceedings of the Special Interest Group on Computer Science Education (SIGCSE)*. [S.l.], 2013. Citado 3 vezes nas páginas 30, 35 e 36.

LANSIQUOT, R. D.; CABO, C. Strategies to integrate writing in problem-solving courses: Promoting learning transfer in an interdisciplinary context. In: AMERICAN SOCIETY FOR ENGINEERING EDUCATION, SEATTLE, WA, UNITED STATES. *Proceedings of the American Society for Engineering Education (ASEE)*. [S.l.], 2015. Citado 4 vezes nas páginas 30, 31, 35 e 36.

LISHINSKI, A. et al. The influence of problem solving abilities on students performance on different assessment tasks in cs1. In: ACM MEMPHIS, TN, UNITED STATES. *Proceedings of the Special Interest Group on Computer Science Education (SIGCSE)*. [S.l.], 2016. Citado 5 vezes nas páginas 19, 31, 35, 36 e 38.

LU, I. R. R.; THOMAS, D. R.; ZUMBO, B. D. Embedding irt in structural equation models: A comparison with regression based on irt scores. *Structural Equation Modeling*, Taylor & Francis, v. 12, n. 2, 2005. Citado na página 25.

LUXTON-REILLY, A. et al. Ladebug: An online tool to help novice programmers improve their debugging skills. In: ACM LARNACA, CYPRUS. *Proceedings of the Innovation and Technology in Computer Science Education (ITiCSE)*. [S.l.], 2018. Citado na página 15.

LÓPEZ, P. E. M. et al. The gobstones method for teaching computer programming. In: IEEE. *Proceedings of the Latin American Computer Conference (CLEI)*. [S.l.], 2017. Citado 3 vezes nas páginas 19, 31 e 36.

MALIK, S. I. Enhancing practice and achievement in introductory programming using an adri editor. In: IEEE, BANGKOK, THAILAND. *Proceedings of the IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*. [S.l.], 2016. Citado 3 vezes nas páginas 31, 35 e 36.

MALIK, S. I. et al. Learning problem solving skills: Comparison of e-learning and m-learning in an introductory programming course. *Education and Information Technologies*, Springer, v. 24, n. 5, 2019. Citado 3 vezes nas páginas 31, 35 e 36.

MALIK, S. I.; NEILSON, J. C. Impact of a new teaching and learning approach in an introductory programming course. *Journal of Educational Computing Research*, v. 55, n. 6, 2017. Citado 3 vezes nas páginas 31, 35 e 36.

MATHEW, R.; MALIK, S. I.; TAWAFK, R. M. Teaching problem solving skills using an educational game in a computer programming course. *Informatics in Education*, v. 18, n. 2, 2019. Citado 3 vezes nas páginas 18, 31 e 36.

MCCRACKEN, M. et al. A multi-national, multi-institutional study of assessment of programming skills of first-year cs students. In: ACM. *Working group reports from ITiCSE on Innovation and technology in computer science education*. [S.l.], 2001. p. 125–180. Citado na página 15.

MEDEIROS, R. P.; RAMALHO, G. L.; FALCÃO, T. P. A systematic literature review on teaching and learning introductory programming in higher education. *IEEE Transactions on Education*, IEEE, v. 62, n. 2, p. 77–90, 2018. Citado na página 15.

MLADENOVIĆ, M.; KRPAN, D.; MLADENOVIĆ, S. Learning programming from scratch. In: *Proceedings of the International Conference on New Horizons in Education*. [S.l.: s.n.], 2017. Citado 4 vezes nas páginas 19, 31, 35 e 36.

MULLER, O.; BUTMAN, A.; BUTMAN, M. Opening a (sliding) window to advanced topics. In: ACM BOLOGNA, ITALY. *Proceedings of the ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE)*. [S.l.], 2017. Citado 3 vezes nas páginas 31, 35 e 36.

NIEMELÄ, P. et al. Computational thinking as an emergent learning trajectory of mathematics. In: ACM KOLI, FINLAND. *Proceedings of the Koli Calling International Conference on Computing Education Research*. [S.l.], 2017. Citado 2 vezes nas páginas 31 e 36.

PALMER, J. D.; FLIEGER, J.; HILLENBRAND, E. Javagrinder: A web-based platform for teaching early computing skills. In: AMERICAN SOCIETY FOR ENGINEERING EDUCATION, SEATTLE, WA, UNITED STATES. *Proceedings of the American Society for Engineering Education (ASEE)*. [S.l.], 2011. Citado 3 vezes nas páginas 31, 35 e 36.

PAPADOPOULOS, Y.; TEGOS, S. Using microworlds to introduce programming to novices. In: IEEE. *Proceedings of the Panhellenic Conference on Informatics*. [S.l.], 2012. Citado 4 vezes nas páginas 31, 35, 36 e 37.

PARK, C. J.; HYUN, J. S. Effects of abstract thinking and familiarity with programming languages on computer programming ability in high schools. In: IEEE, WELLINGTON, NEW ZEALAND. *Proceedings of the International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*. [S.l.], 2014. Citado 5 vezes nas páginas 19, 31, 36, 37 e 38.

PARK, C. J.; HYUN, J. S.; HEUILAN, J. Effects of gender and abstract thinking factors on adolescents' computer program learning. In: IEEE, EL PASO, TX, UNITED STATES. *Proceedings of the Frontiers in Education Conference (FIE)*. [S.l.], 2015. Citado 5 vezes nas páginas 19, 31, 36, 37 e 38.

PASQUALI, L. *Psicometria: Teoria dos Testes na Psicologia e na Educação*. [S.l.]: Editora Vozes Limitada, 2017. Citado 3 vezes nas páginas 21, 23 e 25.

PHILIP, M.; RENUMOL, V. G.; GOPEEKRISHNAN, R. A pragmatic approach to develop computational thinking skills in novices in computing education. In: IEEE. *Proceedings of the IEEE International Conference in MOOC, Innovation and Technology in Education (MITE)*. [S.l.], 2013. Citado 4 vezes nas páginas 31, 35, 36 e 37.

PRIMI, R. Psicometria: Fundamentos matemáticos da teoria clássica dos testes. *Avaliação Psicológica*, Instituto Brasileiro de Avaliação Psicológica, v. 11, n. 2, 2012. Citado 3 vezes nas páginas 20, 21 e 22.

RAPKIEWICZ, C. E. et al. Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais. *RENOTE: revista novas tecnologias na educação [recurso eletrônico]*. Porto Alegre, RS, 2007. Citado na página 14.

ROSE, S. Bricolage programming and problem solving ability in young children: An exploratory study. In: UNIVERSITY OF THE WEST OF SCOTLAND, PAISLEY, SCOTLAND. *Proceedings of the European Conference on Games Based Learning*. [S.l.], 2016. Citado 3 vezes nas páginas 31, 36 e 37.

SANDS, P. Addressing cognitive load in the computer science classroom. *ACM Inroads*, ACM New York, NY, United States, v. 10, n. 1, 2019. Citado 5 vezes nas páginas 14, 15, 31, 35 e 36.

SILVA, I.; SILVA, I. M.; SANTOS, M. S. Análise de problemas e soluções aplicadas ao ensino de disciplinas introdutórias de programação. *Universidade Federal Rural de Pernambuco, Recife-PE*, 2009. Citado na página 14.

SKALKA, J.; DRLÍK, M. Educational model for improving programming skills based on conceptual microlearning framework. In: SPRINGER, CHAM. *Proceedings of the International Conference on Interactive Collaborative Learning (ICL)*. [S.l.], 2018. Citado 5 vezes nas páginas 19, 31, 35, 36 e 37.

SMETSERS, R. W.; SMETSERS, S. Problem solving and algorithmic development with flowcharts. In: ACM NIJMEGEN, NETHERLANDS. *Proceedings of the Workshop in Primary and Secondary Computing Education (WiPSCE)*. [S.l.], 2017. Citado 4 vezes nas páginas 14, 31, 36 e 37.

SOUZA, L. M. et al. Mathematics and programming: Marriage or divorce? In: IEEE, LIMA, PERU. *Proceedings of the World Conference on Engineering Education (EDUNINE)*. [S.l.], 2019. Citado 5 vezes nas páginas 20, 31, 36, 37 e 38.

SPANGSBERG, T. H.; FINCHER, S.; DZIALLAS, S. Non-traditional novices' perceptions of learning to program: A framework of developing mental models. In: IEEE, SAN JOSE, CA, UNITED STATES. *Proceedings of the Frontiers in Education Conference (FIE)*. [S.l.], 2018. Citado 4 vezes nas páginas 31, 36, 37 e 38.

STATTER, D.; ARMONI, M. Teaching abstraction in computer science to 7th grade students. *ACM Transactions on Computing Education (TOCE)*, ACM New York, NY, United States, v. 20, n. 1, 2020. Citado 4 vezes nas páginas 19, 31, 36 e 37.

TAHERI, S. M.; SASAKI, M.; NGETHA, H. T. Evaluating the effectiveness of problem solving techniques and tools in programming. In: IEEE, LOGON, UNITED KINGDOM. *Proceedings of the Science and Information Conference (SAI)*. [S.l.], 2015. Citado 6 vezes nas páginas 14, 18, 19, 31, 36 e 37.

TEW, A. E.; MCCRACKEN, W. M.; GUZDIAL, M. Impact of alternative introductory courses on programming concept understanding. In: ACM. *Proceedings of the first international workshop on Computing education research*. [S.l.], 2005. p. 25–35. Citado na página 15.

THEVATHAYAN, C.; HAMILTON, M. Supporting diverse novice programming cohorts through flexible and incremental visual constructivist pathways. In: ACM VILNIUS, LITHUANIA. *Proceedings of the Innovation and Technology in Computer Science Education (ITiCSE)*. [S.l.], 2015. Citado 4 vezes nas páginas 18, 31, 36 e 37.

TOPALLI, D.; CAGILTAY, N. E. Improving programming skills in engineering education through problem-based game projects with scratch. *Computers Education*, Elsevier, v. 120, 2018. Citado 3 vezes nas páginas 31, 35 e 36.

VEERASAMY, A. K. et al. Relationship between perceived problem-solving skills and academic performance of novice learners in introductory programming courses. *Journal of Computer Assisted Learning*, John Wiley Sons Ltd, v. 35, n. 2, 2019. Citado 4 vezes nas páginas 31, 35, 36 e 38.

WATSON, C.; LI, F. W. Failure rates in introductory programming revisited. In: ACM. *Proceedings of the 2014 conference on Innovation & technology in computer science education*. [S.l.], 2014. p. 39–44. Citado na página 14.

ÇOKLAR, A. N.; AKCAY, A. Evaluating programming self-efficacy in the context of inquiry skills and problem-solving skills: A perspective from teacher education. *World Journal on Educational Technology: Current Issues*, Science Park Research, Organization and Counseling, v. 10, n. 3, 2018. Citado 2 vezes nas páginas 31 e 36.

ANEXO A – TERMO DE ASSENTIMENTO

Caro Responsável/Representante legal,

Convidamos o menor a participar do projeto de pesquisa “**Mensuração de habilidades cognitivas introdutórias de programação por meio de uma avaliação adaptativa informatizada**” coordenado pelos professores Dr. Wilkerson de Lucena Andrade e Dr. João Arthur Brunet Monteiro vinculados ao Programa de Pós Graduação da Universidade Federal de Campina Grande, a quem poderar contatar/consultar a qualquer momento que julgar necessário através dos e-mails {wilkerson, joao.arthur}@computacao.ufcg.edu.br. O documento abaixo contém todas as informações necessárias sobre a pesquisa que estamos fazendo. A colaboração do menor neste estudo será de muita importância para nós, mas se desistir a qualquer momento, isso não causará nenhum prejuízo a ele.

Eu,, portador da Cédula de identidade, RG, e inscrito no CPF....., abaixo assinado(a), concordo de livre e espontânea vontade em autorizar a participação do menor como voluntário(a) deste estudo. Declaro que obtive todas as informações necessárias, bem como todos os eventuais esclarecimentos quanto às dúvidas por mim apresentadas.

Estou ciente que:

- Este estudo tem por objetivo mensurar as habilidades cognitivas introdutórias relacionadas à programação de alunos em cursos técnicos e superiores com intuito de analisar suas limitações e descrever métodos para aprimorar o processo de ensino-aprendizagem.
- O menor será submetido aos seguintes procedimentos: i) fornecer respostas à exercícios durante a aplicação de instrumentos psicométricos; e, ii) avaliar o seu nível de satisfação ao usar os instrumentos psicométricos. Na qual serei beneficiado por avaliar minhas habilidades introdutórias em Programação.
- Caso o menor sinta riscos por está intimidado(a) ou desconfortado(a) durante a participação na pesquisa, poderá desistir a qualquer momento, retirando seu consentimento, sem que isso lhe traga nenhum prejuízo ou penalidade e receberá apoio necessário a questão apresentada.
- Todas as informações obtidas serão sigilosas e o nome do menor não será identificado em nenhum momento. Os dados serão guardados em local seguro e a divulgação dos resultados será feita de maneira que não permita a identificação dele.
- Se o menor tiver algum gasto decorrente a participação na pesquisa, será ressarcido, caso solicite. Em qualquer momento, se o menor sofrer algum dano comprovadamente decorrente desta pesquisa, será indenizado.
- Caso o menor se sinta prejudicado (a) por participar desta pesquisa, poderei recorrer ao Comitê de Ética em Pesquisas com Seres Humanos – CEP, do Hospital Universitário Alcides Carneiro - HUAC, situado a Rua: Dr. Carlos Chagas, s/ n, São José, CEP: 58401 – 490, Campina Grande-PB, Tel: 2101 – 5545, E-mail: cep@huac.ufcg.edu.br; Conselho Regional de Medicina da Paraíba e a Delegacia Regional de Campina Grande.
- Atesto recebimento de uma via assinada deste Termo de Assentimento, conforme recomendações da Comissão Nacional de Ética em Pesquisa (CONEP). Outros esclarecimentos sobre esta pesquisa, poderei entrar em contato com o pesquisador principal Me. Jucelio Soares dos Santos, Rua Severino Pimentel através do e-mail jucelio@copin.ufcg.edu.br.

Campina Grande - PB, de de 20

(Assinatura do responsável ou representante legal)

(Testemunha 1 | Nome/RG/Telefone)

(Testemunha 2 | Nome/RG/Telefone)

**Me. Jucelio Soares dos Santos
(Responsável pelo projeto)**

ANEXO B – TERMO DE CONSENTIMENTO LIVRE ESCLARECIDO

Você está sendo convidado(a) a participar do projeto de pesquisa “**Mensuração de habilidades cognitivas introdutórias de programação por meio de uma avaliação adaptativa informatizada**” coordenado pelos professores Dr. Wilkerson de Lucena Andrade e Dr. João Arthur Brunet Monteiro vinculados ao Programa de Pós Graduação da Universidade Federal de Campina Grande, a quem poderar contactar/consultar a qualquer momento que julgar necessário através dos e-mails {wilkerson, joao.arthur}@computacao.ufcg.edu.br. O documento abaixo contém todas as informações necessárias sobre a pesquisa que estamos fazendo. Sua colaboração neste estudo será de muita importância para nós, mas se desistir a qualquer momento, isso não causará nenhum prejuízo a você.

Eu,, portador da Cédula de identidade, RG, e inscrito no CPF....., abaixo assinado(a), concordo de livre e espontânea vontade em participar como voluntário(a) deste estudo. Declaro que obtive todas as informações necessárias, bem como todos os eventuais esclarecimentos quanto às dúvidas por mim apresentadas.

Estou ciente que:

- Este estudo tem por objetivo mensurar as habilidades cognitivas introdutórias relacionadas à programação de alunos em cursos técnicos e superiores com intuito de analisar suas limitações e descrever métodos para aprimorar o processo de ensino-aprendizagem.
- Serei submetido aos seguintes procedimentos: i) fornecer respostas à exercícios durante a aplicação de instrumentos psicométricos; e, ii) avaliar o seu nível de satisfação ao usar os instrumentos psicométricos. Na qual serei beneficiado por avaliar minhas habilidades introdutórias em Programação.
- Caso sinta riscos por está intimidado(a) ou desconfortado(a) durante a participação na pesquisa, poderei desistir a qualquer momento, retirando meu consentimento, sem que isso me traga nenhum prejuízo ou penalidade e receberei apoio necessário a questão apresentada.
- Todas as informações obtidas serão sigilosas e meu nome não será identificado em nenhum momento. Os dados serão guardados em local seguro e a divulgação dos resultados será feita de maneira que não permita a minha identificação.
- Se eu tiver algum gasto decorrente de minha participação na pesquisa, serei ressarcido, caso solicite. Em qualquer momento, se eu sofrer algum dano comprovadamente decorrente desta pesquisa, serei indenizado.
- Caso me sinta prejudicado (a) por participar desta pesquisa, poderei recorrer ao Comitê de Ética em Pesquisas com Seres Humanos – CEP, do Hospital Universitário Alcides Carneiro - HUAC, situado a Rua: Dr. Carlos Chagas, s/ n, São José, CEP: 58401 – 490, Campina Grande-PB, Tel: 2101 – 5545, E-mail: cep@huac.ufcg.edu.br; Conselho Regional de Medicina da Paraíba e a Delegacia Regional de Campina Grande.
- Atesto recebimento de uma via assinada deste Termo de Consentimento Livre e Esclarecido, conforme recomendações da Comissão Nacional de Ética em Pesquisa (CONEP). Outros esclarecimentos sobre esta pesquisa, poderei entrar em contato com o pesquisador principal Me. Jucelio Soares dos Santos através do e-mail: jucelio@copin.ufcg.edu.br.

Campina Grande - PB, de de 20

(Assinatura do participante)

(Testemunha 1 | Nome/RG/Telefone)

(Testemunha 2 | Nome/RG/Telefone)

Me. Jucelio Soares dos Santos
(Responsável pelo projeto)