



**UNIVERSIDADE ESTADUAL DA PARAÍBA  
CAMPUS VII - PATOS  
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS  
CURSO DE GRADUAÇÃO EM BACHARELADO EM COMPUTAÇÃO**

**ROGÉRIO MIRANDA CORREIA LIMA FILHO**

**AVALIAÇÃO ADAPTATIVA DE UM SISTEMA WEB PARA MENSURAR  
HABILIDADES COGNITIVAS PREDITORAS EM NOVATOS EM PROGRAMAÇÃO**

**PATOS - PB**

**2022**

ROGÉRIO MIRANDA CORREIA LIMA FILHO

**AVALIAÇÃO ADAPTATIVA DE UM SISTEMA WEB PARA MENSURAR  
HABILIDADES COGNITIVAS PREDITORAS EM NOVATOS EM PROGRAMAÇÃO**

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

**Orientador:** MSc. Jucelio Soares dos Santos

**PATOS - 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.

L732a Lima Filho, Rogerio Miranda Correia.  
Avaliação adaptativa de um sistema web para mensurar habilidades cognitivas preditoras em novatos em programação [manuscrito] / Rogerio Miranda Correia Lima Filho. - 2022.  
48 p. : il. colorido.

Digitado.

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

"Orientação : Prof. Me. Jucélio Soares dos Santos, Departamento de Computação - CCT."

1. Programação. 2. Habilidades preditoras. 3. Testes adaptativos informatizados. 4. Algoritmos. 5. Teoria de resposta ao ítem. I. Título

21. ed. CDD 005.1

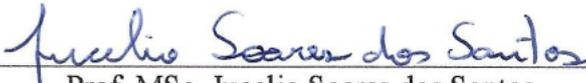
ROGÉRIO MIRANDA CORREIA LIMA FILHO

AVALIAÇÃO ADAPTATIVA DE UM SISTEMA WEB PARA MENSURAR HABILIDADES  
COGNITIVAS PREDITORAS EM NOVATOS EM PROGRAMAÇÃO

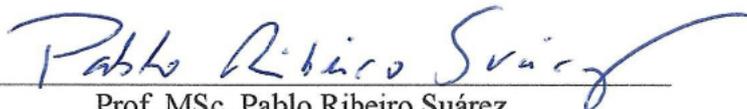
Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Computação do Centro de Ciências Exatas e Sociais Aplicadas da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de bacharel em Computação.

Trabalho aprovado em 01/08/2022.

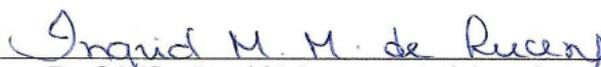
**BANCA EXAMINADORA**

  
\_\_\_\_\_

Prof. MSc. Jucelio Soares dos Santos  
(Orientador)

  
\_\_\_\_\_

Prof. MSc. Pablo Ribeiro Suárez  
(Examinador)

  
\_\_\_\_\_

Prof. MSc. Ingrid Morgane Medeiros de Lucena  
(Examinadora)

Dedico este trabalho a todas as grandes amizades que espero levar pra minha vida e àquele, em especial, que sempre esteve do meu lado nos momentos mais difíceis.

## **AGRADECIMENTOS**

A meus pais, que nunca mediram esforços para assegurar minha educação como prioridade, por proporcionarem a melhor vida que podiam me dar e por incentivarem meus sonhos.

Aos meus amigos, que me acompanham, me incentivam diante das dificuldades e entenderam minha ausência enquanto produzo este trabalho.

Aos meus professores, pelos seus ensinamentos e tempo disponibilizado a mim, permitindo minha evolução como estudante e profissional.

*"Few are those who see with their  
own eyes and feel with their own hearts."*

**Albert Einstein**

## RESUMO

Esta pesquisa tem como objetivo principal integralizar a seleção adaptativa em um instrumento informatizado, a fim de melhorar o processo de mensuração das habilidades preditoras de programação. Para tanto, elaboraremos algoritmos baseado na Teoria de Resposta ao Item para seleção adaptativa dos itens, definindo para tanto um critério para a habilidade inicial, procedimento de seleção dos itens, método de estimação da habilidade e diferentes critérios de parada. Em seguida, compararemos os algoritmos por meio de uma base de dados dos respondentes a fim de verificar o desempenho em relação ao tempo e a estimativa da habilidade. Como resultados, entre os algoritmos desenvolvidos para selecionar itens de forma adaptativa, o algoritmo que obteve melhor precisão no processo de estimação das habilidades possui as seguintes características: i) máxima verossimilhança como método de estimação da habilidade; ii) máxima informação de fisher como método de seleção adaptativa; iii) obter um controle de exposição de itens; iv) e para o método de parada do instrumento, uma combinação de fatores como o erro padrão na estimação  $<0,01$  e a quantidade mínima e máxima de itens administrados. Com base nas características do algoritmo anterior, o tempo de resposta aos itens diminuiu consideravelmente para todas as habilidades investigadas.

**Palavras-chaves:** Testes Adaptativos Informatizados. Algoritmos de Seleção Adaptativa. Teoria de Resposta ao Item. Habilidades Preditoras de Programação.

## ABSTRACT

This research aims to integrate adaptive selection into a computerized instrument to improve the measuring process of predictive programming skills. To do so, we will develop algorithms based on Item Response Theory for adaptive items selection, defining a criterion for initial skill, item selection procedure, skill estimation method, and different stopping criteria. Next, we will compare the algorithms through a database of respondents to verify performance against time and skill estimation. As a result, among the algorithms developed to adaptively select items, the algorithm that obtained the best precision in the skill estimation process has the following characteristics: i) maximum likelihood as a skill estimation method; ii) maximum fisher information as an adaptive selection method; iii) obtain an item exposure control; iv) and for the instrument stop method, a factors combination such as the standard error in the estimation  $<0.01$  and the minimum and maximum quantity of administered items. Based on the characteristics of the previous algorithm, the response time to items decreased considerably for all investigated skills.

**Keywords:** Computerized Adaptive Tests. Adaptive Selection Algorithms. Theory Item Response. Predictive Programming Skills.

## LISTA DE ILUSTRAÇÕES

Figura 1 – CCI's exemplo . . . . .	22
Figura 2 – Fluxograma para elaborar um TAI . . . . .	22
Figura 3 – FII's exemplo . . . . .	26
Figura 4 – Critérios para Elaboração de Algoritmos para Seleção Adaptativa de Itens. .	31
Figura 5 – Boxplot de frequência de itens aplicados em todas as habilidades para o Z3.	32
Figura 6 – Estimação das Habilidades entre os Algoritmos de Seleção Adaptativa dos Itens. . . . .	34
Figura 7 – Tempo de Respostas ao Itens por Habilidade entre os Algoritmos. . . . .	37

## LISTA DE TABELAS

Tabela 1 – Número Mínimo e Máximo de Itens para todas as Habilidades. . . . .	33
Tabela 2 – Estatística Descritiva da Estimativa de Habilidades entre os Algoritmos. . .	35
Tabela 3 – Teste de Normalidade de Estimativas de Habilidades entre os Algoritmos. .	35
Tabela 4 – Correlação da Estimação das Habilidades entre a EAP e os Algoritmos Adaptativos. . . . .	35
Tabela 5 – Estatística Descritiva no Tempo de Resposta aos Itens por Habilidade entre os Algoritmos. . . . .	38
Tabela 6 – Teste de Normalidade do Tempo de Resposta aos Itens por Habilidade entre os Algoritmos. . . . .	38
Tabela 7 – Análise do Tempo de Resposta nas Habilidades entre os Algoritmos. . . . .	39

## **LISTA DE ABREVIATURAS E SIGLAS**

3ML	Modelo Logístico de 3 parâmetros
CCI	Curva Característica do Item
CS1	Curso Introdutório de Programação
EAP	Expected A Posteriori
FII	Função de Informação do Item
TAI	Teste Adaptativo Informatizado
TCT	Teoria Clássica dos Testes
TRI	Teoria de Resposta ao Item

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> . . . . .	<b>13</b>
<b>1.1</b>	<b>Cenário Técnico Científico</b> . . . . .	<b>13</b>
<b>1.2</b>	<b>Problema</b> . . . . .	<b>14</b>
<b>1.3</b>	<b>Objetivos</b> . . . . .	<b>15</b>
<b>1.4</b>	<b>Questão de Pesquisa</b> . . . . .	<b>15</b>
<b>1.5</b>	<b>Estrutura do trabalho</b> . . . . .	<b>15</b>
<b>2</b>	<b>TEMAS E TRABALHOS RELACIONADOS</b> . . . . .	<b>16</b>
<b>2.1</b>	<b>Habilidades Predictoras de Programação</b> . . . . .	<b>16</b>
<i>2.1.1</i>	<i>Resolução de Problemas</i> . . . . .	<i>16</i>
<i>2.1.2</i>	<i>Pensamento Abstrato</i> . . . . .	<i>17</i>
<i>2.1.3</i>	<i>Raciocínio Matemático</i> . . . . .	<i>17</i>
<i>2.1.4</i>	<i>Flexibilidade Cognitiva</i> . . . . .	<i>18</i>
<b>2.2</b>	<b>Teorias de Mensuração</b> . . . . .	<b>18</b>
<i>2.2.1</i>	<i>Teoria Clássica dos Testes</i> . . . . .	<i>19</i>
<i>2.2.1.1</i>	<i>Coeficiente de Ponto Bisserial</i> . . . . .	<i>19</i>
<i>2.2.1.2</i>	<i>Coeficiente Alpha de Cronbach</i> . . . . .	<i>20</i>
<i>2.2.2</i>	<i>Teoria de Resposta ao Item</i> . . . . .	<i>20</i>
<b>2.3</b>	<b>Teste Adaptativo Informatizado</b> . . . . .	<b>22</b>
<i>2.3.1</i>	<i>Método de Estimação</i> . . . . .	<i>24</i>
<i>2.3.2</i>	<i>Método de Seleção Adaptativa</i> . . . . .	<i>25</i>
<i>2.3.3</i>	<i>Método de Controle de Exposição de Itens</i> . . . . .	<i>26</i>
<i>2.3.4</i>	<i>Crítério de Parada do Instrumento</i> . . . . .	<i>27</i>
<b>3</b>	<b>METODOLOGIA</b> . . . . .	<b>28</b>
<b>3.1</b>	<b>Técnica e Métrica</b> . . . . .	<b>28</b>
<b>3.2</b>	<b>Identificação das Variáveis Dependentes e Independentes</b> . . . . .	<b>28</b>
<b>3.3</b>	<b>Participantes</b> . . . . .	<b>28</b>
<b>3.4</b>	<b>Preparação</b> . . . . .	<b>29</b>
<b>3.5</b>	<b>Análise dos dados</b> . . . . .	<b>29</b>
<b>3.6</b>	<b>Análise às Ameaças</b> . . . . .	<b>29</b>
<b>3.7</b>	<b>Execução da Pesquisa</b> . . . . .	<b>30</b>
<b>4</b>	<b>ANÁLISE E RESULTADOS</b> . . . . .	<b>31</b>
<b>4.1</b>	<b>Desenvolvimento de Algoritmos para Seleção Adaptativa de Itens e Estimativa de Habilidades</b> . . . . .	<b>31</b>
<b>4.2</b>	<b>Eficiência de Algoritmos para Seleção Adaptativa de Itens e Estimativa de Habilidade</b> . . . . .	<b>33</b>

4.2.1	<i>Precisão na Estimativa da Habilidade</i> . . . . .	33
4.2.2	<i>Tempo de Aplicação</i> . . . . .	36
5	<b>CONSIDERAÇÕES FINAIS E SUGESTÕES PARA TRABALHOS FUTUROS</b> . . . . .	40
	<b>REFERÊNCIAS</b> . . . . .	41
	<b>ANEXO A – TERMO DE ASSENTIMENTO</b> . . . . .	44
	<b>ANEXO B – TERMO DE CONSENTIMENTO LIVRE ESCLARECIDO</b>	45
	<b>ANEXO C – SCRIPT</b> . . . . .	46

## 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

Os Cursos Introdutórios de Programação (CS1) têm altas taxas de evasão e reprovação, equivalentes a um terço dos alunos matriculados neste curso em todo o mundo (WATSON; LI, 2014). Recentemente, a comunidade científica e acadêmica tem vindo a questionar as metodologias práticas utilizadas no Ensino Superior e Técnico para estimular/medir habilidades nestes cursos (SANDS, 2019). Percebe-se que um grupo de ingressantes em CS1 se destaca mais do que o outro. Reforçando que esses indivíduos possuem diferenças em sua trajetória acadêmica e habilidades ou aptidões prévias que os ajudam a compreender os conceitos introdutórios de programação (ANDRADE, 2022) (DANTAS, 2022).

Essas habilidades que precedem as habilidades introdutórias de programação são conhecidas como habilidades preditivas, a saber: Resolução de Problemas<sup>1</sup>, Pensamento Abstrato<sup>2</sup>, Raciocínio Matemático<sup>3</sup> e Flexibilidade Cognitiva<sup>4</sup>(ANDRADE, 2022) (DANTAS, 2022). Tais habilidades podem ser estimuladas/medidas antes do CS1 pelo professor a fim de conhecer o nível de habilidades de seus alunos para as disciplinas do primeiro semestre, de forma que eles lecionem suas aulas cientes das habilidades preditivas que os alunos possuem maior domínio ou não (ANDRADE, 2022).

Pensando nisso, em parceria com o Laboratório de Neuropsicologia Cognitiva e Inovação Tecnológica da Universidade Federal de Campina Grande, o Laboratório de Práticas de Software e a Universidade Estadual da Paraíba, foi desenvolvido o Screening Programming<sup>5</sup> destinado a rastrear habilidades preditoras em programação (ANDRADE, 2022) (DANTAS, 2022). Trata-se de um sistema que apoia os educadores de cursos de Ensino Superior e Técnico da área de informática a mensurar habilidades preditoras de programação (ANDRADE, 2022).

Resultados anteriores mostram que o Screening Programming apresenta itens que contemplam as habilidades preditoras de programação. Tais itens possuem uma boa análise de conteúdo, boas propriedades psicométricas e boa confiabilidade em avaliações profissionais (DANTAS, 2022). Após a instrumentalização, os usuários avaliaram a sua usabilidade e obtiveram uma empatia pelo rastreamento/evolução das habilidades preditoras de programação feito por meio da versão informatizada (ANDRADE, 2022).

<sup>1</sup> Capacidade do aluno de encontrar uma solução para um determinado problema;

<sup>2</sup> Capacidade do aluno de dividir um determinado problema em partes menores e resolvê-lo separadamente;

<sup>3</sup> Capacidade do aluno de se sentir confortável com a matemática, ajudando-o a resolver um problema;

<sup>4</sup> Capacidade do aluno de pensar em soluções diferentes para o mesmo problema.

<sup>5</sup> Disponível em: <https://www.screeningprogramming.com/>

## 1.2 Problema

O Screening Programming mensura as habilidades preditoras em programação por meio de itens apresentados sequencialmente. Entretanto, a sequência/quantidade de itens pode não ser adequada para o nível de habilidade em que o aluno se encontra. Um item com um nível de dificuldade baixo para sujeitos com alta habilidade se torna desgastante, uma vez que as respostas corretas a esses itens acrescentam pouca informação para a avaliação da proficiência desses indivíduos (PRIMI, 2012). Outrossim, itens que não oferecem nenhum desafio podem deixar o indivíduo entediado, respondendo sem maiores cuidados. Da mesma forma, itens com um nível de dificuldade alto para sujeitos com baixa proficiência se tornam desgastantes, propiciando que os indivíduos respondam de forma aleatória e incorporando respostas incorretas nos itens, de tal forma que não oferecem muita informação às estimativas, tal fato pode influenciar na desistência do aluno na prática de exercícios (SOUZA et al., 2019). Sendo assim, os itens apresentados sequencialmente podem não discriminar bem os alunos com baixa/alta habilidade, um item mal escolhido pode superestimar ou subestimar a habilidade do indivíduo, influenciando diretamente no resultado do aluno no instrumento.

Esta realidade poderia ser melhorada se o Screening Programming fornecesse uma escolha dos itens presentes em seu banco, de acordo com o nível de habilidade do aluno. A ideia é que essas mudanças reduzam o número de itens a serem administrados na avaliação e estimem a habilidade com melhor precisão. Em contrapartida, ocorre que, o Screening Programming estimula a habilidade do indivíduo utilizando a Teoria Clássica dos Testes (TCT), que usa como referência o número total de sucessos de um instrumento para avaliar o desempenho dos alunos. Embora simples e direta, essa abordagem tem limitações em relação ao número certo de itens a serem administrados no instrumento, de modo que os indivíduos possam responder a itens que não correspondem ao seu nível de habilidade, causando cansaço e gerando respostas não confiáveis. (WAINER et al., 2000).

Para cobrir essas falhas, é possível utilizar a Teoria de Resposta do Item (TRI) e administrar os itens do Screening Programming por meio de um algoritmo que seleciona de forma adaptativa os itens de acordo com o nível de habilidade do sujeito. A TRI considera o item como a unidade básica de análise e procura representar a probabilidade de um indivíduo responder ao item de acordo com os seus parâmetros e sua característica latente. O Screening Programming adaptativo pode ajudar os profissionais do campo a resolver problemas que exigem ferramentas de qualidade, além de conhecimento especializado. Neles, o mesmo conjunto de itens é aplicado a todos os indivíduos, permitindo uma maior precisão, velocidade, facilidade de atualização e, além disso, esse instrumento é menos sujeito a erros na divulgação de seus resultados (SARTES; SOUZA-FORMIGONI, 2013).

Dessa forma, esse problema motivou a realização desta pesquisa que visa contribuir em uma solução mais próxima e pertinente para uma das realidades singulares da área de Educação em Computação. É preciso melhorar o processo de avaliação dos alunos previamente em CS1, a fim de apoiar o professor no planejamento e execução de atividades de ensino inovadoras, bem

como promover oportunidades contextualizadas/personalizadas para a aprendizagem dos alunos nos níveis técnico e superior.

### **1.3 Objetivos**

A presente pesquisa tem como objetivo principal integralizar a seleção adaptativa em um instrumento informatizado, a fim de melhorar o processo de mensuração das habilidades preditoras de programação. Para alcançar o objetivo geral desta pesquisa, foram necessários atingir os seguintes objetivos específicos:

- Elaborar algoritmos para seleção adaptativa dos itens, definindo para tanto um critério para a habilidade inicial, procedimento de seleção dos itens, método de estimação da habilidade e critérios de paradas;
- Comparar os algoritmos por meio de uma base de dados dos respondentes a fim de verificar o desempenho em relação a estimativa da habilidade e ao tempo de resposta.

### **1.4 Questão de Pesquisa**

Considerando as falhas no processo de medição de Screening Programming para habilidades preditoras de programação, acreditamos ser importante investir na integração da seleção adaptativa por meio de algoritmos que melhorem o desempenho em relação ao tempo de aplicação dos itens, bem como na estimativa de a habilidade dos alunos. Neste contexto, respondemos às seguintes questões de investigação:

- **QP1.** Qual algoritmo adaptativo tem melhor precisão na estimativa de habilidades preditoras de programação?
- **QP2.** O tempo de resposta aos itens usando o algoritmo adaptativo é menor quando comparado ao método tradicional?

### **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, analisamos 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

As habilidades preditivas são habilidades específicas que antecedem um marco ou o desenvolvimento de outras habilidades mais estruturadas, neste caso, habilidades que antecedem as habilidades de programação são: Resolução de Problemas, Pensamento Abstrato, Raciocínio Matemático e Flexibilidade Cognitiva.

#### 2.1.1 *Resolução de Problemas*

Os alunos entram em CS1 com uma ampla gama de níveis de habilidade, principalmente em seu conhecimento prévio de programação e habilidades de resolução de problemas (SANDS, 2019). Ensinar conceitos de programação e pensamento computacional é uma tarefa desafiadora, principalmente quando os alunos têm limitações na resolução de problemas (BASNET et al., 2018).

A resolução de problemas é um conceito útil que descreve o esforço consciente no processamento controlado de informações e visa identificar, descobrir ou inventar uma solução para um problema (ALSHAYE; TASIR; JUMAAT, 2019).

As habilidades de resolução de problemas são essenciais para a compreensão de programação para iniciantes em CS1 (CHAUDHRY; RASOOL, 2012). Porém, o professor aborda essa habilidade somente nas primeiras lições desses cursos. Já os livros didáticos relevantes incluem apenas em algumas lições ou capítulos iniciais (SANDERS et al., 2013).

Apesar disso, muitas instituições utilizam ferramentas a fim de introduzir conceitos de programação para iniciantes (HOOSHYAR et al., 2015a). Outras pesquisas nesta área revelaram que as limitações nas habilidades de resolução de problemas são intensificadas pelas complexidades associadas ao ambiente de desenvolvimento e à sintaxe da linguagem adotada pelos cursos (HOOSHYAR et al., 2015b) (HOOSHYAR et al., 2015c).

Alunos com maiores habilidades de resolução de problemas acham a programação fácil e podem dominar a programação com pouca ou nenhuma dificuldade, independentemente do ambiente de programação. Em contraste, os alunos com poucas habilidades de resolução de problemas acham a programação difícil de entender e muitas vezes são incapazes de dominá-la. E os alunos que se enquadram entre as duas categorias? Este é provavelmente o grupo mais representado; são alunos que dominam a programação mas com algumas dificuldades (MLADENOVÍĆ; KR PAN; MLADENOVÍĆ, 2017).

### **2.1.2 *Pensamento Abstrato***

O pensamento abstrato afeta a forma como as pessoas assimilam informações, elaboram julgamentos e regulam o comportamento. Tendo como exemplo, um programa que faz uso de estruturas de iteração, o pensamento abstrato pode influenciar diretamente a compreensão deste programa. Se o nível de pensamento abstrato do aluno for mais elevado, ele compreenderá melhor o programa com esse tipo de estrutura. No momento em que o professor fornece ao aluno um problema simples de programação, a habilidade de pensamento abstrato não é um fator decisivo para a avaliação dos alunos. Todavia, quando a complexidade do problema é maior, o alto nível de capacidade de pensamento abstrato exerce um papel essencial na solução do problema (PARK; HYUN, 2014).

Há um efeito moderador entre a familiaridade com programação e a capacidade de pensamento abstrato (PARK; HYUN, 2014) (PARK; HYUN; HEUILAN, 2015). Quando os alunos têm altos níveis de pensamento abstrato, suas habilidades de programação se destacam entre os outros. Cada vez que os professores desenvolvem o nível de pensamento abstrato dos alunos, a distância psicológica do tempo também aumenta. A familiaridade é um fator chave na compreensão dos programas, como resultado, os alunos aprenderão mais e praticarão mais sobre linguagens de programação.

### **2.1.3 *Raciocínio Matemático***

As habilidades de programação geralmente estão relacionadas às habilidades de resolução de problemas. Certos tipos de habilidades matemáticas afetam as habilidades analíticas. Essas habilidades ajudam a aprender a experiência de programação em um computador. Além da sintaxe e semântica das linguagens de programação, pensamento lógico e matemático, habilidades numéricas e de visualização, conhecimento de álgebra e cálculo e o conhecimento de condições e funções recursivas também são essenciais para projetar funções e programas em atividades de programação (ATTALLAH; ILAGURE; CHANG, 2018).

Alguns conceitos matemáticos, como indução, inferência e raciocínio lógico, são essenciais para a programação de computadores. Além disso, compreender estruturas, símbolos, operadores, fórmulas, regras e formas matemáticas e lógicas pode agregar valor às sequências de instruções definidas pelo programador e evitar erros e enganos no código (DRACHOVA et al., 2015).

Para escrever um código totalmente especificado e verificável, os alunos de computação precisam ser capazes de fazer inferências matemáticas sobre os componentes de software e seus relacionamentos, precisam modelá-los por meio de construções matemáticas e precisam usar linguagem matemática precisa para compreender e escrever especificações e declarações formais com a utilização da linguagem matemática precisa (DRACHOVA et al., 2015). Existe uma correlação entre habilidades matemáticas e potencial de programação. Os alunos consideram a lógica de programação e os métodos de ensino a principal fonte de dificuldades (SOUZA et al.,

2019). Entre eles, estão justamente os conceitos matemáticos citados.

#### **2.1.4 Flexibilidade Cognitiva**

Flexibilidade cognitiva é a capacidade de guiar pensamentos e classificar soluções divergentes na resolução de problemas, analisando soluções menos óbvias para um problema (DURAK, 2018). Analisando este conceito, podemos ter em mente diversas estratégias comportamentais em uma certa situação, obtendo diferentes alternativas para um mesmo objetivo. Com uma variedade de alternativas, a busca por soluções mais descomplicadas e eficazes torna-se mais simples, por consequência, a compreensão dos problemas torna-se mais clara.

A flexibilidade cognitiva permite que os alunos adotem as estratégias de aprendizagem mais eficazes relacionadas ao tópico de pesquisa, ou determinem as etapas para resolver problemas, encontrar soluções, controlar o processo de aprendizagem e produtos ou oportunidades autorreguláveis. Portanto, o professor deve desenvolver essa habilidade para melhorar o desempenho dos alunos em programação (DURAK, 2020).

Entretanto, essa habilidade não se adequa para certos grupos de pessoas, especificamente, em indivíduos com baixo nível de adaptabilidade. Temos como principal exemplo, os portadores do Transtorno do Espectro Autista, que de forma geral, são dependentes de uma rotina fixa. Essa característica dificulta a adaptação dos mesmos às mudanças, fazendo com que o conceito da flexibilidade cognitiva não possa ser aplicado.

Para potenciar essa capacidade, os professores devem adotar outros métodos que permitam aos alunos melhorar o seu desempenho. Outro ponto relacionado é que o desenvolvimento da flexibilidade cognitiva depende da prática e assimilação do conhecimento das etapas que a compõem (DURAK, 2020). Logo, estar aberto a outros pontos de vista, incentivar a dinâmica de grupo dentro da sala de aula, promover o debate entre diferentes opiniões e ativar a criatividade dos alunos são algumas etapas que auxiliam de forma positiva para uma flexibilidade cognitiva consistente, com um raciocínio mais claro e preciso.

## **2.2 Teorias de Mensuração**

Utilizar números nos dá permissão para quantificar os fenômenos naturais analisados em estudos científicos. Quando atrelado ao método científico, esta quantificação faz uso da instrumentação e das técnicas de medição, possibilitando a compreensão destes fenômenos. Na pesquisa em Educação e Computação, almejamos métodos para a coleta de dados e a avaliação cognitiva dos estudantes em relação aos nossos tópicos de pesquisa científica (ARAÚJO et al., 2019).

A Teoria Clássica dos Testes (TCT) e a Teoria de Resposta ao Item (TRI) são dois referenciais utilizados para construção, validação e avaliação de instrumentos em Psicologia e em Educação (PASQUALI, 2017). Dentro do contexto educacional, tanto o TCT quanto o TRI, podemos aplicá-las na avaliação de construtos cognitivos.

### 2.2.1 Teoria Clássica dos Testes

A TCT tem como principal característica utilizar o escore, ou pontuação total de um instrumento como medida avaliativa de seu desempenho. Esta pontuação total consiste na soma do “escore verdadeiro” do indivíduo com os possíveis erros de medida cometidos durante a avaliação (PRIMI, 2012).

O escore verdadeiro, de forma resumida, seria a medição perfeita do conhecimento da pessoa, sem qualquer erro ao acaso ou distratores internos ou externos; sem qualquer possibilidade de erro vindo do instrumento de avaliação com o mesmo medindo unicamente através da habilidade que está sendo analisada; à grosso modo, uma medição sem nenhum viés (HUTZ; BANDEIRA; TRENTINI, 2015).

Outro aspecto desta teoria é que ela faz uso de normas para interpretar as pontuações de um instrumento. Essas normas tornam-se nossa principal referência para a interpretação e classificação das pontuações. Podemos citar como situação-exemplo, quando situamos a posição de um indivíduo no construto medido pelo instrumento em questão, ou comparar a pontuação de dois indivíduos analisados (ARAÚJO et al., 2019).

Por meio da pontuação, utilizamos algumas medidas para ponderar a qualidade dos itens analisados e do instrumento de avaliação. Temos como principais exemplos, o *coeficiente de correlação do ponto bisserial* e o *coeficiente alfa de Cronbach*.

#### 2.2.1.1 Coeficiente de Ponto Bisserial

O coeficiente ponto bisserial é usado para indicar quais itens do instrumento são determinantes para um melhor resultado (PASQUALI, 2017). O coeficiente ponto bisserial pode ser calculado entre duas variáveis em testes tradicionais: uma variável numérica e outra nominal categórica. No caso em questão, a variável categórica é uma variável dicotômica, ou seja, que possui apenas dois valores possíveis (certo/errado).

Para calcular a correlação entre essas duas variáveis, dada a normalidade da amostra, trate o coeficiente de Pearson da maneira usual, que é denominado coeficiente de correlação de ponto bisserial (BAKER; KIM, 2017).

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 indica os itens que têm influência mais significativa na habilidade estimada; se o examinado acertar este item, terá boas chances de passar no exame. O

coeficiente de ponto bisserial varia entre -1 e +1, e quanto mais próximo de 1, mais discriminante será o item, ou seja, mais forte é a correlação deste item com o escore total. Esse valor mostrará que aquele item é essencial para o instrumento.

### 2.2.1.2 Coeficiente Alpha de Cronbach

O Coeficiente Alpha de Cronbach indica a confiabilidade do instrumento, ou seja, examina a homogeneidade dos itens que compõem o instrumento, analisando a magnitude das relações entre os itens e o escore total. Podemos calcular o Coeficiente Alpha de Cronbach 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.

O Coeficiente Alpha de Cronbach varia de 0 a 1; quanto mais próximo de 1 indica que o instrumento tem boa consistência interna (ANDRADE; LAROS; GOUVEIA, 2010). 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).

### 2.2.2 Teoria de Resposta ao Item

Enquanto a TCT verifica a probabilidade de o examinado acertar um determinado item por acaso, com a TRI essa possibilidade pode ser identificada por meio de respostas a outros itens (consistência nas respostas). Com a TRI também é possível estabelecer uma comparação entre indivíduos de uma mesma população que realizaram testes diferentes, ou mesmo uma comparação entre populações através de testes com alguns itens em comum (ANDRADE; TAVARES; VALLE, 2000).

Com a TRI, podemos elaborar instrumentos de avaliação educacional, calibrar itens, estimar a habilidade dos examinados e verificar outros processos de desenvolvimento de instrumentos. A TRI fundamenta o ajuste dos dados ao modelo logístico adotado. Desta forma, frequentemente, examinados diferentes ou até o mesmo examinado podem ter suas proficiências comparadas aos itens do instrumento, devido a utilização de parâmetros que determinam a estimativa da habilidade, independentemente da amostra utilizada (BAKER, 2001).

Assim sendo, por meio de um conjunto de fatores ou variáveis hipotéticas pode-se prever o comportamento de um examinado ao um item do instrumento, representado por meio de uma

função matemática monótona crescente cujo gráfico é denominado Curva Característica do Item (CCI) (PASQUALI, 2017).

A CCI é um gráfico que representa a probabilidade de um examinado em acertar o item (BAKER, 2001) (ARAÚJO et al., 2019). Cada item possui uma CCI que determina a sua qualidade. A TRI influencia as CCI 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,

$\theta$  representa o traço/habilidade latente de um indivíduo;

$a_j$  representa o parâmetro de discriminação do item;

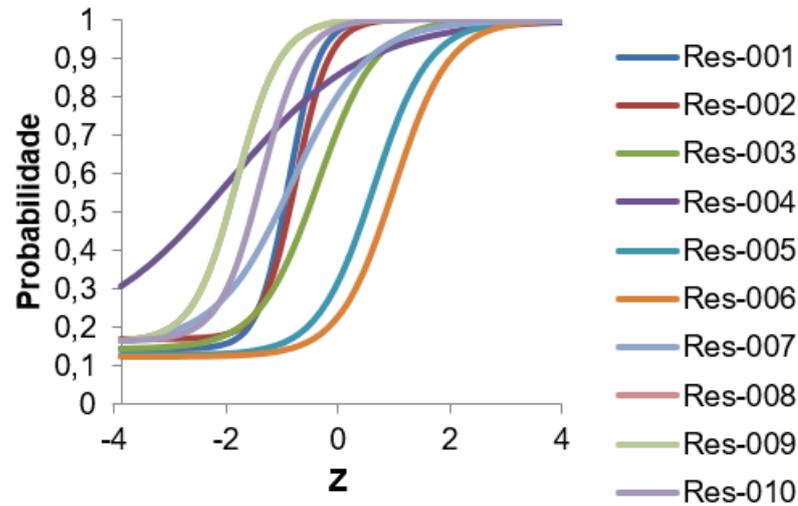
$b_j$  representa o parâmetro de dificuldade do item;

$c_j$  representa o parâmetro de acertar o item pelo acaso;

A discriminação do item (parâmetro  $a$ ) descreve quantos examinados 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).

A seguir, apresentamos na Figura 1 um exemplo de gráfico que plota algumas CCI's. Este gráfico mostra dez 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 entre -4 a +4. O eixo  $y$  representa a probabilidade do item de resposta correta variando de 0 a 1. As curvas 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.

Figura 1 – CCI's exemplo



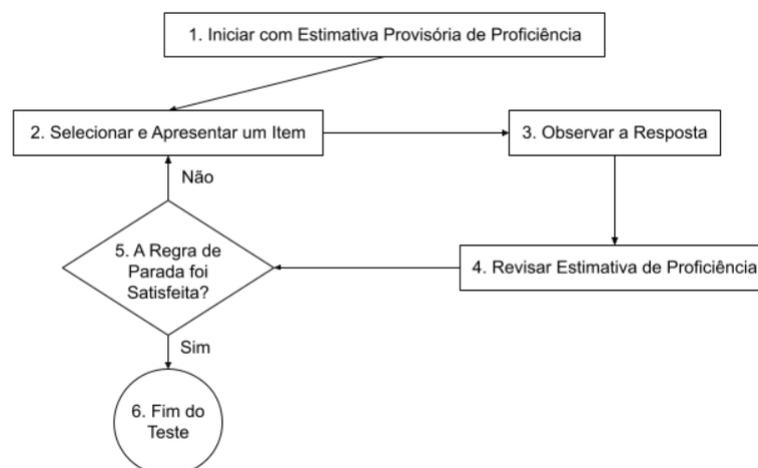
Fonte: (DANTAS, 2022).

No exemplo da Figura 1, o item Res-004 é o menos discriminativo e 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 menor chance de acertar pelo acaso. Já o item Res-002 é o item mais fácil de acertar pelo acaso.

### 2.3 Teste Adaptativo Informatizado

Resumidamente, um TAI tem a seguinte estrutura: i) O algoritmo adaptativo entrega um item e o examinando responde; ii) O algoritmo adaptativo reestimar a proficiência do examinado com base nessa resposta. Após calcular essa estimativa, iii) o algoritmo adaptativo selecionará um novo item. A lógica para a construção de testes adaptativos está ilustrada na Figura 2.

Figura 2 – Fluxograma para elaborar um TAI



Fonte: Criada pelo autor (2022).

TAI busca ajustar os itens do instrumento ao nível de habilidade de cada sujeito examinado, da mesma forma que um especialista intuitivamente faria (COSTA, 2009). Como resultado, esses testes oferecem várias vantagens em relação aos testes tradicionais, como:

- *Redução em um tamanho de teste.* O número do item em TAI é menor que um teste tradicional para o mesmo nível de precisão. Essa redução só é possível devido à redução das informações sobre a estimativa de habilidade em cada item administrado;
- *Flexibilidade nas baterias de teste.* Assim como nos testes tradicionais, é desnecessária a participação de todos os sujeitos ao mesmo tempo na aplicação do teste;
- *Maior controle das regras de teste.* Por ser informatizado, o instrumento consegue obter uma melhor controle sobre as regras de aplicação do instrumento;
- *Maior controle na correção dos testes.* Reduz erros que podem ocorrer em processos de correção óptica;
- *Melhora a usabilidade do teste.* Testes computadorizados motivam mais os indivíduos porque utilizam recursos multimídia que os tornam mais atrativos do que os testes tradicionais.

Por limitação, esses testes exigem mais recursos financeiros e humanos quando comparados às versões tradicionais de papel e lápis. Tais recursos visam construir o banco de dados de itens e segurança da informação que requerem investimentos em hardware e software necessários para sua criação e aplicação. A sua construção é um processo de várias etapas, necessitando do apoio de uma equipe multidisciplinar para orientar todas as etapas. Essa equipe pode ser composta por especialistas na habilidade analisada, estatísticos e psicólogos psicométricos. Quando o instrumento é informatizado, como o TAI, também precisamos de profissionais de informática.

Na prática, a estimativa da proficiência dos examinados em um teste convencional (papel-e-lápis) é feita por meio de alguns softwares comerciais, como o Bilog-MG. Os programas em TAI, por sua vez, são desenvolvidos especificamente para um determinado teste, utilizando ferramentas proprietárias, que na maioria das vezes não se aplicam a outros testes. Essa prática ocorre porque, diferentemente dos testes tradicionais, os TAI's podem incluir diferentes métodos para estimar as proficiências.

Um TAI ajusta dinamicamente os itens que melhor avaliam uma determinada proficiência. Basicamente quatro mecanismos podem ser definidos para a estimativa eficiente dessa medida:

- Método de Estimação;
- Método de Seleção Adaptativa;
- Método de Controle de Exposição de Itens;
- Critério de Parada do Instrumento.

### 2.3.1 Método de Estimação

O TAI estima as proficiências dos examinados iterativamente após a resposta de cada item no instrumento. Na TRI, existe vários estimadores para se obter a medida de proficiência dos examinados. Os mais difundidos são os Bayesianos e o da Máxima Verossimilhança.

Entre os métodos bayesianos, um dos utilizados é Expected A Posteriori (EAP) (KOLEN; TONG, 2010) (LU; THOMAS; ZUMBO, 2005). Cada examinado recebe uma pontuação que melhor identifica a habilidade na escala. O significado do termo “a posteriori” deriva do conceito bayesiano de probabilidade posterior; Ou seja, é uma distribuição da probabilidade posterior de pontuações em relação ao traço latente. Esta previsão de pontuações para um determinado caso depende da resposta padrão do item e dos parâmetros estimados no 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,

$\theta_s$  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 Máxima Verossimilhança estima a capacidade de um examinando. É um procedimento iterativo, como no caso da estimação de parâmetros de itens. Começa com algum valor a priori para a habilidade do examinado e os valores conhecidos dos parâmetros do item. O procedimento utiliza esses dados para calcular a probabilidade de acerto de cada item. O procedimento então obtém um ajuste para a estimativa de habilidade que melhora de acordo com as probabilidades calculadas a partir da resposta do item dada durante o teste. O procedimento repete esse processo até que o ajuste se torne pequeno o suficiente para que a mudança na capacidade estimada seja insignificante, resultando em uma estimativa da capacidade do examinando. Este processo é então repetido separadamente para cada examinando no teste (BAKER; KIM, 2004). A Equação 2.5 define a estimativa do traço latente.

$$\theta_{s+1} = \theta_s + \frac{\sum_{i=1}^n a_i [u_i - P_i \theta_s]}{\sum_{i=1}^n a_i^2 P_i(\theta_s) Q_i(\theta_s)} \quad (2.5)$$

Onde,

$\theta_s$  representa a capacidade estimada do examinado dentro de  $s$  iterações;

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

$u_i$  é a resposta do examinando ao item (1 quando certo e 0 quando errado);

$P_i(\theta_s)$  é a probabilidade de acerto do item  $i$ , segundo a CCI em  $\theta$ ;

$Q_i(\theta_s)$  é a probabilidade de resposta incorreta ao item  $i$ , segundo a CCI em  $\theta$ .

Infelizmente, não há como saber a habilidade real do examinado. Neste caso, o melhor procedimento é estimá-lo. No entanto, deve-se obter um erro padrão da habilidade estimada que

indica acurácia. O examinando, hipoteticamente, ao fazer o mesmo teste várias vezes, assumindo que não há itens de testes anteriores, terá sua capacidade  $\theta$  estimada pelo instrumento cada vez que os aplicadores o administrarem. Erro padrão mede a variabilidade dos valores de  $\theta$  em torno do valor da habilidade desconhecida do sujeito. Este erro é estimado usando a Equação 2.6.

$$SE(\theta) = \frac{1}{\sqrt{\sum_{i=1}^n a_i^2 P_i(\theta_s) Q_i(\theta_s)}} \quad (2.6)$$

É interessante notar que o termo sob o sinal da raiz quadrada é precisamente o denominador da Equação 2.5. Como resultado, o erro padrão estimado resulta do subproduto da estimativa de habilidade do examinando.

### 2.3.2 Método de Seleção Adaptativa

Em um TAI, um dos componentes mais importantes é selecionar itens em todo o instrumento. O instrumento é mais eficiente em estimar a habilidade do examinando quando possui itens que não são muito difíceis nem muito fáceis. No entanto, os métodos de seleção adaptativa de itens não avaliam apenas o parâmetro de dificuldade dos itens, mas também encontram uma medida de informação em busca de uma melhor escolha de itens para estimar a proficiência do examinando. Essas informações são uma combinação dos parâmetros do item e da estimativa de proficiência. Nesta seção, apresentaremos um método para estimar a habilidade do examinando, trata-se do **Crítério de Máxima Informação de Fisher**.

A Máxima Informação de Fisher (FII) gera um gráfico que contém informações de um item em todos os pontos ao longo do continuum do traço latente (PASQUALI, 2017); ou seja, permite saber não apenas quanta informação um item acumula em um determinado  $\theta$ , mas também em que  $\theta$  o item tem a maior quantidade de informações. A Máxima Informação de Fisher tem sido o método de análise de item mais comumente usados pelos construtores de teste (BAKER, 2001). A Equação 2.7 define a FII:

$$I(\theta) = a^2 \frac{Q_i(\theta)(P_i(\theta) - c)^2}{P_i(\theta)(1 - c)^2} \quad (2.7)$$

Onde,

$a$  representa o parâmetro de discriminação do item;

$c$  representa o parâmetro dificuldade do item;

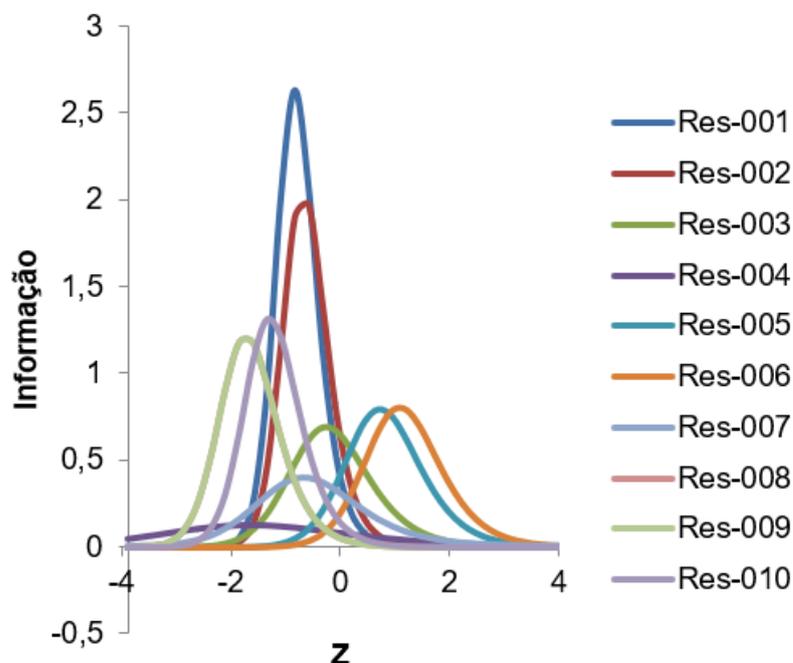
$P_i(\theta)$  é a probabilidade de acerto do item  $i$ , segundo a CCI em  $\theta$ ;

$Q_i(\theta)$  é a probabilidade de resposta incorreta ao item  $i$ , segundo a CCI em  $\theta$ .

A seguir, apresentamos na Figura 3 um exemplo de gráfico que plota algumas funções da Máxima Informação de Fisher. O gráfico da Figura 3 mostra dez funções, cada um correspondendo a um item, modelado a partir da Equação 2.7, que usaremos como exemplos para ilustrar o comportamento dos itens. O valor da habilidade  $Z$  varia -4 a +4 (BAKER, 2001). O eixo  $y$

representa a área onde a habilidade é calculada. As linhas indicam quanta informação cada item fornece em uma determinada região de traço latente.

Figura 3 – FII's exemplo



Fonte: Criada pelo autor (2022).

No exemplo da Figura 2, é possível perceber o quanto de informação cada item fornece em uma determinada região do traço latente para atividades do instrumento. Por exemplo, o item Res-002 oferece mais informação para avaliar sujeitos de habilidade mediana, ou seja, e um Teste Adaptativo Informatizado (TAI), esse item seria o primeiro item a ser escolhido, se o instrumento situar a habilidade Theta em 0 como a habilidade média.

### 2.3.3 Método de Controle de Exposição de Itens

Muitos TAIs encontram uma base para a seleção de itens em procedimentos estatísticos e impõem restrições ao procedimento de seleção de itens. Essas restrições destinam-se a controlar atributos específicos, como equilíbrio de conteúdo ou frequência de exposição do item.

A imposição de restrições torna-se necessária para melhor aproveitamento das estruturas presentes nos bancos de itens. A ideia principal na implementação de algoritmos é realizar um TAI com as especificações exatas (e a mesma validade) como um teste padrão de papel e lápis e ainda fornecer menos itens.

Esta seção apresentará duas estruturas que o algoritmo pode aplicar: controlar a frequência de exposição dos itens e balancear o conteúdo. A restrição à frequência de exposição do item é significativa no TAI, pois ao utilizar o critério de Máxima Informação, por exemplo, os itens com maior informação tendem a ser administrados várias vezes no TAI, o que pode levar muitos

examinandos a memorizá-los, acrescentando assim um erro na estimativa de proficiência e, conseqüentemente, prejudicando a validade do instrumento.

A restrição de balanceamento de conteúdo permite dividir o banco de itens em várias seções, cada uma das quais representará um conteúdo (também conhecido como habilidade, competência, descritor) que se deseja avaliar no TAI. Dessa forma, o TAI conterà uma boa variedade de itens de diferentes competências da mesma forma que no instrumento de papel e lápis.

#### **2.3.4 Critério de Parada do Instrumento**

Uma característica essencial do TAI são os critérios de parada que dependem dos objetivos do instrumento. Instrumentos podem ser usados em i) seleção/classificação ou ii) mensuração da habilidade.

Para fins de classificação, o algoritmo compara a proficiência de um examinado com algum valor de corte. Nesse caso, o algoritmo pode usar tanto a estimativa de proficiência quanto o erro padrão associado da medida como critérios de parada. O algoritmo classifica o indivíduo, verificando se está acima ou abaixo do valor de corte. Este valor de corte varia dentro da escala do traço latente, entre -4 e +4.

Após decidir o ponto de corte, o algoritmo encerra o instrumento quando esta condição é satisfeita. O resultado de cada instrumento será um conjunto de avaliações feitas por um grupo de examinandos que tenham uma taxa de erro de pelo menos 5%. O algoritmo controla a taxa de erro alterando o tamanho do intervalo de confiança do erro padrão da medida em torno da estimativa de proficiência.

Uma regra de parada diferente se aplica quando o objetivo do TAI não é para classificação. Nesse caso, é desejável medir cada examinando com um nível fixo de precisão; ou seja, um nível predeterminado do erro padrão é fixado. Essa decisão resultará em um conjunto de medidas de "equilíbrio", onde todos os participantes do teste pontuarão com precisão equivalente. O TAI permite que os usuários especifiquem o nível de medição de erro padrão que é desejável para cada examinando implementar uma medição precisa. Assumindo que o banco de itens possui um número suficiente de itens distribuídos em toda a faixa de traços latentes e que o tamanho do teste é suficiente para cada examinado, o algoritmo encerra o instrumento quando o nível de erro padrão da medida é satisfeito.

Em alguns TAIs, o administrador encerra o algoritmo quando atinge um número fixo de itens ou impõe um tempo limite. No caso de instrumentos de classificação, essa decisão pode afetar a estimativa de alguns examinandos. No outro caso, o término prematuro do TAI não resultará em medição precisa, pois a medida de erro padrão não diminui para todos os examinandos na mesma proporção. Assim, deve-se evitar usar o limite de tempo e o tamanho fixo do instrumento como critério de parada em um instrumento.

### **3 METODOLOGIA**

Neste capítulo, apresentamos o planejamento dos estudos conduzidos para contemplar o objetivo desta pesquisa. Ao final desta pesquisa, queremos saber se integralização da seleção adaptativa ao Screening Programming por meio de algoritmos melhora o seu desempenho em relação ao tempo de aplicação dos itens, bem como na estimativa da habilidade dos alunos.

#### **3.1 Técnica e Métrica**

Conforme discutimos no Capítulo anterior, o TAI é administrado por um algoritmo tem como requisitos a existência de um banco de itens calibrado. Além disso, este algoritmo possuirá: i) Método de estimação; ii) Método de seleção adaptativa; iii) Método de controle de exposição de itens; e, iv) Método de parada do instrumento.

Com base na literatura, elaboraremos algoritmos adaptativos para seleção de itens no TAI. Conduziremos estudos de simulações. Em todos eles utilizaremos os 40 itens presentes no Screening Programming calibrados pelo 3ML.

#### **3.2 Identificação das Variáveis Dependentes e Independentes**

As Variáveis Independentes a serem utilizadas no estudo são:

- Método de estimação;
- Método de seleção adaptativa;
- Método de controle de exposição de itens;
- Métodos de parada do instrumento.

As Variáveis Dependentes são:

- Estimativa da habilidade;
- Tempo para estimar a habilidade.

#### **3.3 Participantes**

Este estudo foi composto por 100 (cem) participantes, que atenderam aos seguintes critérios de inclusão:

- O participante deve assinar o Termo de Assentimento (Anexo A) ou o Termo de Consentimento Livre Esclarecido (Anexo B) e será informado, de forma simplificada, sobre os procedimentos que serão realizados;

- O participante deverá estar matriculado no primeiro período de um curso de graduação ou curso técnico na área de Ciência da Computação;
- O participante não pode apresentar limitações sensoriais, cognitivas ou queixas auditivas e visuais.

### **3.4 Preparação**

Esta subseção contém informações sobre a organização/estruturação dos elementos considerados para a realização dos estudos de simulação. Utilizamos a base de dados do Screening Programming.

O Screening Programming apresenta 40 itens de múltipla-escolha que avaliam habilidades preditoras de programação (resolução de problemas, pensamento abstrato, raciocínio matemático e flexibilidade cognitiva). Cada habilidade possui cerca de 10 itens. Este estudo considerou todo o banco atual. Sua aplicação foi realizada de forma virtual utilizando o próprio sistema web.

### **3.5 Análise dos dados**

Os dados coletados no Screening Programming foram simulados por meio de script (Anexo C) e analisados com base na estatística descritiva e inferencial, visando atender aos objetivos propostos deste estudo.

### **3.6 Análise às Ameaças**

Consideramos alguns fatores que podem gerar ameaças e influenciar diretamente nas conclusões deste trabalho. Entre eles:

- Usamos o banco de dados do Screening Programming, que pode ter problemas relacionados à interpretação incorreta das perguntas. No entanto, essa ameaça foi mitigada devido à alta consistência nas respostas obtidas pelos respondentes. Além disso, caso os participantes da pesquisa se sentissem intimidados ou desconfortáveis em realizar os testes, o estudo anterior (DANTAS, 2022) aplicou as diretrizes do comitê de ética em pesquisa para minimizar essa possível restrição. O Comitê de Ética em Pesquisa com Seres Humanos da Universidade Federal de Campina Grande e da Universidade Estadual da Paraíba aprovou esta pesquisa (Protocolos: 23933919.4.0000.5182 | 23933919.4.3001.5187). Participaram deste estudo apenas os participantes que assinaram o Termo de Consentimento Livre e Esclarecido (Anexo A) ou o Termo de Consentimento Livre e Esclarecido (Anexo B);
- Os instrumentos foram corrigidos virtualmente para mitigar possíveis erros humanos;
- Como toda pesquisa empírica, este trabalho apresenta ameaças à validade. O número de sujeitos participantes do estudo não permite a generalização dos resultados;

- A amostra deve ser considerada, pois permitiu a formação de um banco de dados baseado em probabilidade, estatística, axiomas de medição e considerando o objetivo do instrumento. Este banco tem o controle centralizado do aplicativo.

### **3.7 Execução da Pesquisa**

Adotamos as seguintes etapas para identificar se a integração da seleção adaptativa no Screening Programming melhorou o processo de estimativa de habilidades:

- Desenvolvemos os algoritmos para seleção adaptativa de itens. Esses algoritmos tiveram as mesmas inicializações, os mesmos procedimentos de seleção de itens e métodos para estimar a habilidade. No entanto, com diferentes critérios de parada;
- Com base nas respostas dos alunos ao banco de itens do Screening Programming, simulamos os TAIs por meio de um script (Anexo C) criado pelos pesquisadores utilizando a linguagem Java;
- Comparamos os algoritmos para verificar seu desempenho em termos de estimativa de habilidade e tempo de resposta.

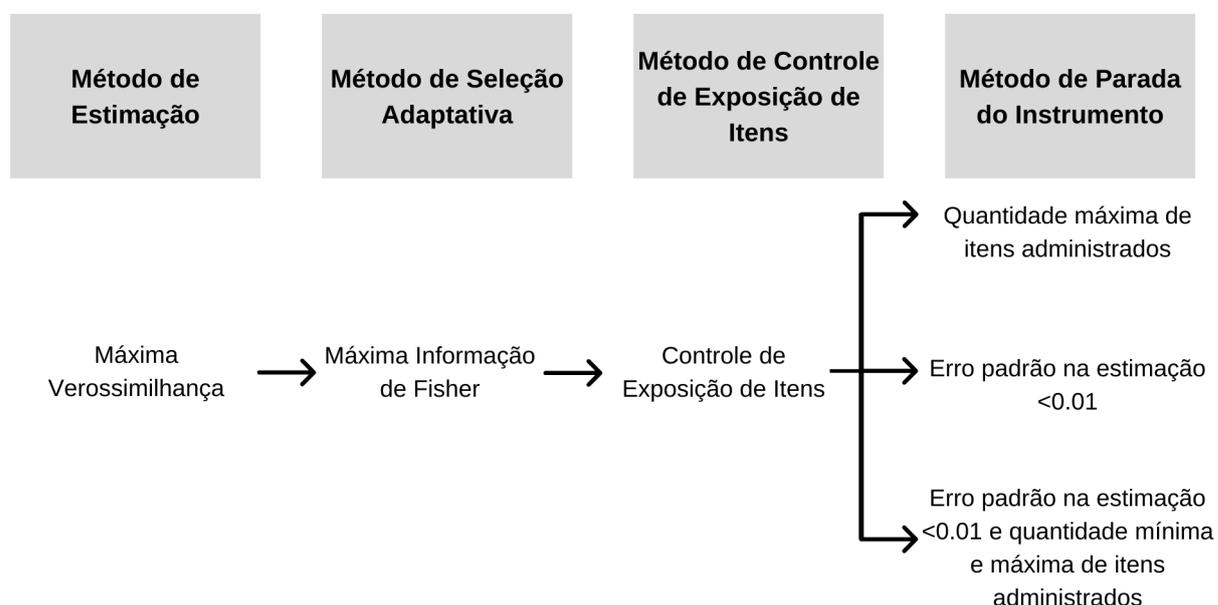
## 4 ANÁLISE E RESULTADOS

Neste capítulo, apresentamos e discutimos os resultados desta pesquisa que visa melhorar o processo de mensuração de habilidades predictoras de programação em um sistema web. Desenvolvemos um TAI que seleciona de forma adaptativa os itens do banco para obter uma precisão na estimativa de habilidade e gerenciar menos itens do que o sistema tradicional.

### 4.1 Desenvolvimento de Algoritmos para Seleção Adaptativa de Itens e Estimativa de Habilidades

Um TAI é gerenciado de acordo com um algoritmo de seleção adaptável. Esse algoritmo especifica os itens em que o indivíduo responderá e sua ordem de apresentação. Conforme discutido nos Capítulos 2 e 3, o algoritmo é composto por: i) Método de Estimação; ii) Método de Seleção Adaptativa; iii) Método de Controle de Exposição do Item; e, iv) Método de Parada do Instrumento. Na Figura 4, apresentamos os critérios adotados para o desenvolvimento de algoritmos de seleção adaptativa de itens.

Figura 4 – Critérios para Elaboração de Algoritmos para Seleção Adaptativa de Itens.



Fonte: Elaborada pelo autor (2022).

Adotamos o método da Máxima Verossimilhança (BAKER; KIM, 2004) como procedimento para estimação das habilidades dos indivíduos. Para seleção dos itens, escolhemos o método da Máxima Informação de Fisher (PASQUALI, 2017). Controlamos a exposição dos itens por meio de seleção própria, uma vez que o item é administrado ao sujeito, ele não será selecionado novamente em uma próxima chamada de itens.

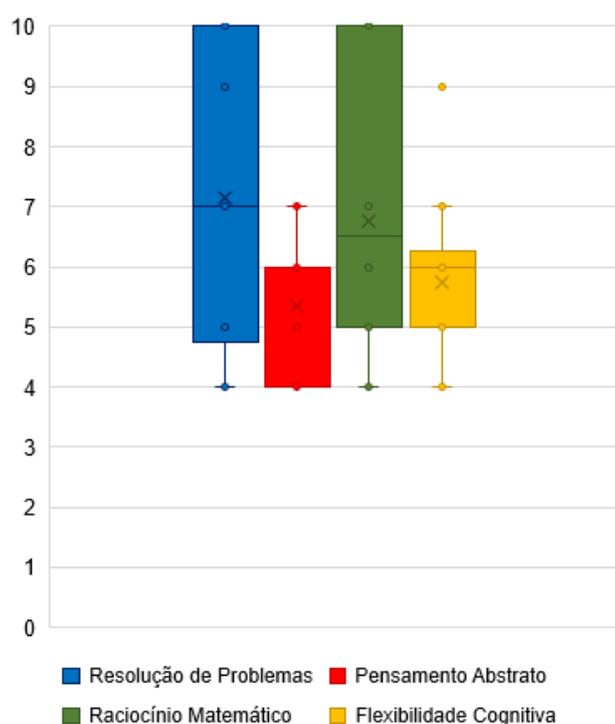
Quando o instrumento termina? É fundamental determinar o momento em que o indivíduo não responderá mais itens (critério de parada). Por fim, variamos os algoritmos apenas no critério de parada do instrumento de três maneiras, a saber:

- **Z2.** quantidade máxima de itens administrados;
- **Z3.** erro padrão da estimação  $<0.01$ ;
- **Z4.** combinação do critério anterior e a quantidade mínima e máxima de itens administrados.

Desenvolvemos um script (Anexo C) para encontrar a estimativa de cada habilidade, o número de itens administrados, bem como o tempo de aplicação, caso o sujeito respondesse ao instrumento adaptativo para cada critério de parada, Z2, Z3 e Z4, respectivamente. Encontramos esses valores simulando a aplicação do instrumento com base nas respostas dadas pelos participantes no pré-teste.

No algoritmo Z2, geramos a estimativa das habilidades dos sujeitos de forma adaptativa, ou seja, todos os itens foram administrados em uma sequência de acordo com o nível de habilidade do sujeito. No algoritmo Z3, adicionamos o erro padrão  $<0,01$  como critério de parada. E com base nos dados obtidos, analisamos que alguns sujeitos responderiam em menor tempo, sem a necessidade de gerenciar todos os itens do banco. Em seguida, obtivemos a frequência de itens aplicados, conforme mostrado na Figura 5 e Tabela 1.

Figura 5 – Boxplot de frequência de itens aplicados em todas as habilidades para o Z3.



Fonte: Elaborada pelo autor (2022).

Tabela 1 – Número Mínimo e Máximo de Itens para todas as Habilidades.

<b>Habilidade</b>	<b>Média</b>	<b>Desvio Padrão</b>	<b>Valor Mínimo</b>	<b>Valor Máximo</b>
Resolução de Problemas	7,136	2,416	5,000	9,750
Pensamento Abstrato	5,364	1,093	4,000	6,000
Raciocínio Matemático	6,773	2,266	5,000	9,250
Flexibilidade Cognitiva	5,727	1,202	5,000	6,000

**Fonte:** Elaborada pelo autor (2022).

Utilizamos os resultados anteriores para gerar o algoritmo com critério de parada Z4, a combinação do erro padrão da estimação ser menor que 0.01 e a quantidade mínima e máxima de itens administrados. Encontramos o valor mínimo e máximo, com base no 1º quartil e 3º quartil, respectivamente.

Conforme a Tabela 1 e a Figura 5, o algoritmo de seleção não selecionará mais itens quando o erro padrão do cálculo da habilidade for menor que 0,01 e o número de itens aplicados for no mínimo 5 e no máximo 10, o que ocorrer primeiro, para a habilidade Resolução de Problemas. Para a habilidade Pensamento Abstrato, entre 4 e 6 itens. Raciocínio Matemático, entre 5 e 10. E, para habilidade Flexibilidade Cognitiva entre 5 e 6 itens.

## 4.2 Eficiência de Algoritmos para Seleção Adaptativa de Itens e Estimativa de Habilidade

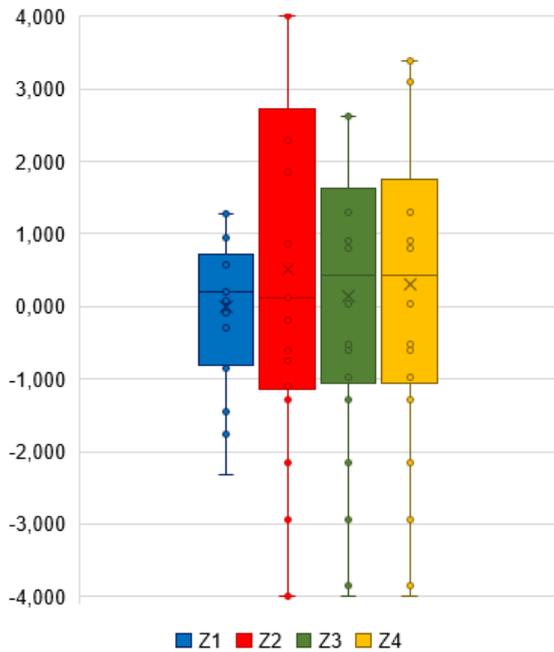
Para verificar a eficiência dos algoritmos desenvolvidos, dimensionamos alguns parâmetros, a saber:

- Precisão na Estimativa de Habilidades;
- Tempo de Aplicação.

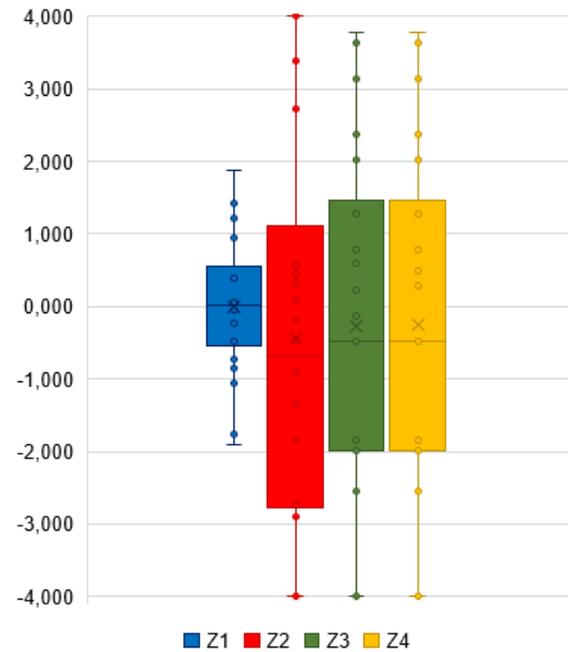
### 4.2.1 Precisão na Estimativa da Habilidade

Estimamos a habilidade dos alunos no pré-teste pela EAP utilizando a macro eirt (**Z1**), bem como a estimativa gerada pelos três algoritmos: **Z2**, **Z3** e **Z4**. As estimativas de habilidade entre os algoritmos são variáveis quantitativas sem causalidade, ou seja, uma variável não influencia a outra. Na Figura 6, apresentamos o boxplot na estimativa da habilidade entre os algoritmos em uma escala de -4 e +4.

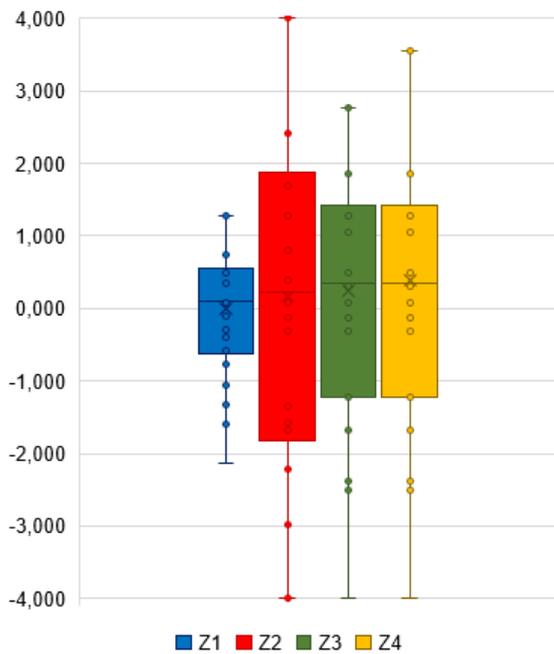
Figura 6 – Estimação das Habilidades entre os Algoritmos de Seleção Adaptativa dos Itens.



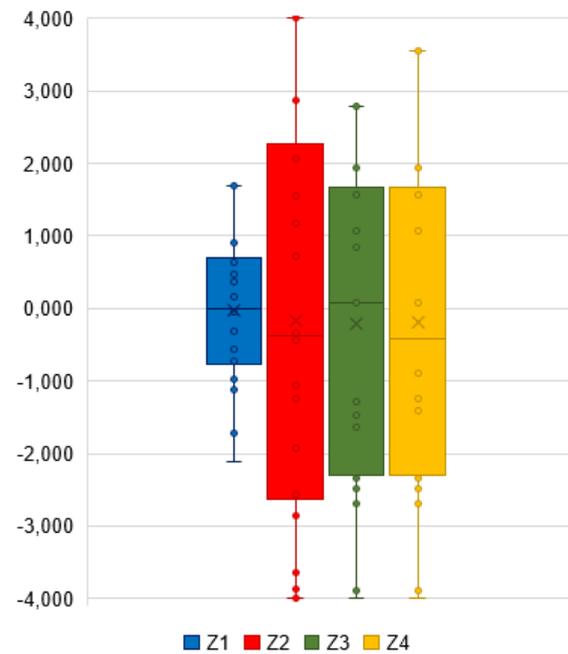
(a) Resolução de Problemas



(b) Pensamento Abstrato



(c) Raciocínio Matemático



(d) Flexibilidade Cognitiva

Fonte: Elaborada pelo autor (2022).

Na Tabela 2, apresentamos a estatística descritiva (média e desvio padrão ( $Sd$ )) da estimativa de habilidades entre os algoritmos. Conforme podemos perceber pelos dados, os algoritmos geraram estimativas diferentes para cada cenário observado.

Tabela 2 – Estatística Descritiva da Estimativa de Habilidades entre os Algoritmos.

Algoritmo	Resolução de Problemas		Pensamento Abstrato		Raciocínio Matemático		Flexibilidade Cognitiva	
	Média	Sd	Média	Sd	Média	Sd	Média	Sd
Z1	<0,001	1,040	<-0,001	0,978	0,001	0,954	-0,015	1,052
Z2	0,506	2,556	-0,430	2,700	0,154	2,520	-0,171	2,743
Z3	0,136	2,042	-0,275	2,292	0,250	1,855	-0,219	2,196
Z4	0,298	2,260	-0,245	2,298	0,393	2,072	-0,184	2,347

Fonte: Elaborada pelo autor (2022).

Para determinar qual dos algoritmos (**Z2**, **Z3** e **Z4**) tem a melhor estimativa quando comparado ao **Z1**, correlacionamos as variáveis. Para isso, primeiramente, calculamos a distribuição normal desses dados através da função *shapiro.test()* presente na linguagem *R*. Na Tabela 3, apresentamos os resultados do teste de normalidade das estimativas de habilidades entre os algoritmos.

Tabela 3 – Teste de Normalidade de Estimativas de Habilidades entre os Algoritmos.

Hipótese Nula	Z1	Z2	Z3	Z4
A estimativa pelo algoritmo na habilidade de Resolução de Problemas não segue uma distribuição normal.	0,083	0,147	0,074	0,158
A estimativa pelo algoritmo na habilidade de Pensamento Abstrato não segue uma distribuição normal.	0,680	0,059	0,303	0,310
A estimativa pelo algoritmo na habilidade de Raciocínio Matemático não segue uma distribuição normal.	0,284	0,246	0,294	0,325
A estimativa pelo algoritmo na habilidade de Flexibilidade Cognitiva não segue uma distribuição normal.	0,763	0,130	0,122	0,254

Fonte: Elaborada pelo autor (2022).

De acordo com os resultados da Tabela 3, todos os dados seguem uma distribuição normal, pois os *p – value* foram maiores que 0,05, refutando assim a hipótese nula e aceitando a hipótese alternativa para todas as habilidades. Dessa forma, calculamos a correlação de Pearson entre as habilidades estimadas no pré-tese pela EAP usando a macro *eirt* (**Z1**) e pelas habilidades estimadas entre os algoritmos adaptativos (**Z2**, **Z3** e **Z4**). Para calcular a correlação de Pearson, usamos a função *cor.test()* presente na linguagem *R*, passando as estimativas para os parâmetros *x* e *y*, e "*pearson*" para o parâmetro *method*. Na Tabela 4, apresentamos esses resultados.

Tabela 4 – Correlação da Estimação das Habilidades entre a EAP e os Algoritmos Adaptativos.

	Habilidade	Z2	Z3	Z4
<b>Z1</b>	Resolução de Problemas	0,893	0,898	0,900
	Pensamento Abstrato	0,910	0,973	0,974
	Raciocínio Matemático	0,801	0,819	0,830
	Flexibilidade Cognitiva	0,958	0,972	0,984

Fonte: Elaborada pelo autor (2022).

Conforme podemos analisar os resultados da Tabela 4, todos os algoritmos apresentam correlação forte ou muito forte para estimar as habilidades dos indivíduos. E, entre as correlações, o algoritmo **Z4** foi o que obteve melhor estimativa para todas as habilidades, respondendo assim a **QP1**. Qual algoritmo adaptativo tem melhor precisão na estimativa de habilidades preditoras de programação?

Com base em nossos resultados, o algoritmo que melhor seleciona itens e estima habilidades com mais precisão, em relação ao teste completo, possui as seguintes características:

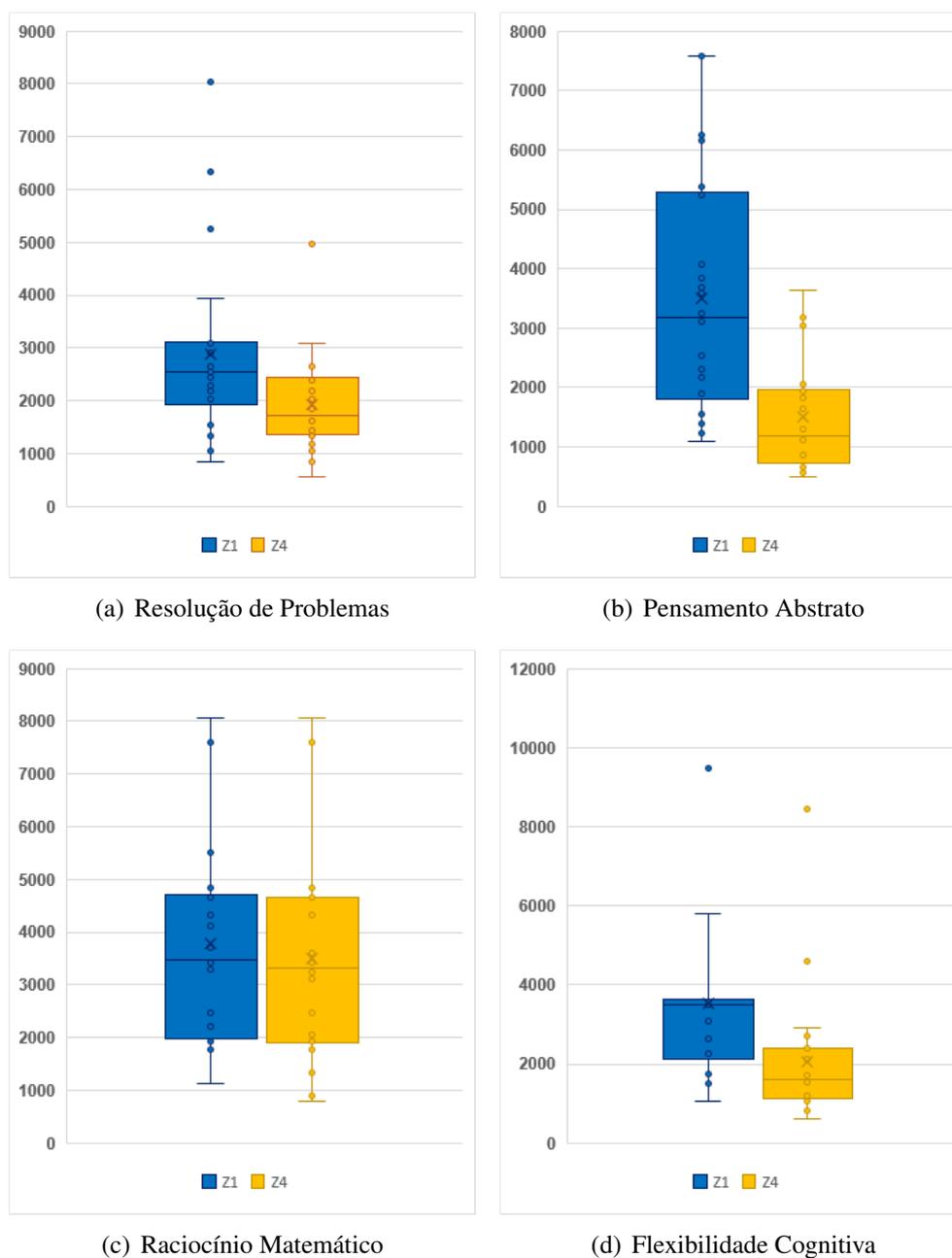
- **Método de Estimação:** máxima verossimilhança;
- **Método de Seleção Adaptativa:** máxima informação de fisher;
- **Método de Controle de Exposição de Itens:** controle de exposição de itens;
- **Método de Parada do Instrumento:** erro padrão na estimação  $<0.01$  e a quantidade mínima e máxima de itens administrados.

#### **4.2.2 Tempo de Aplicação**

O tempo de resposta na aplicação do instrumento depende exclusivamente dos métodos de estimativa de habilidade, seleção dos itens, o controle de exposição dos itens e bem como do método de parada do instrumento. Ou seja, existe uma relação de dependência entre as variáveis. Na Figura 7, apresentamos o boxplot no tempo de resposta aos itens por habilidade entre os algoritmos.

Em Z1, os itens são apresentados sequencialmente para todas as habilidades, independentemente do nível de dificuldade do item e do nível de habilidade do indivíduo. Além disso, é necessário que o indivíduo responda a todos os itens presentes no banco. Em seguida, a habilidade do indivíduo é estimada por meio de alguma ferramenta (por exemplo, a macro eirt), tornando o processo mais demorado. Em Z4, os itens já estão calibrados, exigindo sua administração apenas pelo algoritmo que já demonstrou que gerencia menos itens que o convencional, conforme apresentamos na seção anterior.

Figura 7 – Tempo de Respostas ao Itens por Habilidade entre os Algoritmos.



Fonte: Elaborada pelo autor (2022).

Na Tabela 5, apresentamos a estatística descritiva (média e desvio padrão (*Sd*)) no tempo (em segundos) de resposta aos itens por habilidades entre os algoritmos. Conforme podemos perceber pelos dados, o algoritmo **Z4** administra os itens em menor tempo de resposta. Porém, esse tempo é significativo?

Tabela 5 – Estatística Descritiva no Tempo de Resposta aos Itens por Habilidade entre os Algoritmos.

Algoritmo	Resolução de Problemas		Pensamento Abstrato		Raciocínio Matemático		Flexibilidade Cognitiva	
	$M_d$	$Sd$	$M_d$	$Sd$	$M_d$	$Sd$	$M_d$	$Sd$
Z1	2549	1707,78	3049	1491,1	3476	1412,4	3494	2204,8
Z4	1731	934,35	1189	793,28	3330	1487,2	1622	1686,4

Fonte: Elaborada pelo autor (2022).

Em seguida, verificamos se o tempo de resposta é significativo em relação ao método escolhido. Para tanto, analisamos a normalidade dos dados através da função *shapiro.test()* presente na linguagem *R*. Na Tabela 6, apresentamos os resultados do teste de normalidade para o tempo de resposta aos itens por habilidade entre os algoritmos.

Tabela 6 – Teste de Normalidade do Tempo de Resposta aos Itens por Habilidade entre os Algoritmos.

Hipótese Nula	Z1	Z4
O tempo de resposta aos itens para a habilidade Resolução de Problemas não segue uma distribuição normal.	<0,05	0,013
O tempo de resposta aos itens para a habilidade Pensamento Abstrato não segue uma distribuição normal.	0,039	0,012
O tempo de resposta aos itens para a habilidade Raciocínio Matemático não segue uma distribuição normal.	0,021	0,039
O tempo de resposta aos itens para a habilidade Flexibilidade Cognitiva não segue uma distribuição normal.	<0,05	<0,05

Fonte: Elaborada pelo autor (2022).

De acordo com os resultados da Tabela 6, os dados não seguem uma distribuição normal, pois os *p-value* foram menores que 0,05. Desta forma, usamos o teste de Wilcoxon, com amostras pareadas, a fim de comparar se existe diferença entre as medianas no tempo de respostas em ambas situações. O objetivo do teste dos sinais de Wilcoxon é comparar as performances de cada sujeito (ou pares de sujeitos) no sentido de verificar se existem diferenças significativas entre os seus resultados nas duas situações.

Calculamos o teste de Wilcoxon por meio da função *wilcoxon.test()* presente na linguagem *R*, passando o tempo de resposta para os parâmetros *x* e *y*, e *TRUE* para o parâmetro *paired*. Na Tabela 7, apresentamos esses resultados.

Tabela 7 – Análise do Tempo de Resposta nas Habilidades entre os Algoritmos.

Hipótese Nula	Z1		Z4		Diferença		Wilcoxon <i>p</i> – value
	<i>M<sub>d</sub></i>	<i>IQR</i>	<i>M<sub>d</sub></i>	<i>IQR</i>	<i>M<sub>d</sub></i>	<i>IQR</i>	
Não existe diferença estatística no tempo de resposta aos itens na habilidade Resolução de Problemas em ambos algoritmos.	2548	1006	1730	950	486	1407	0,00048
Não existe diferença estatística no tempo de resposta aos itens na habilidade Pensamento Abstrato em ambos algoritmos.	3176	3050	1188	1158	1736	1531	0,00004
Não existe diferença estatística no tempo de resposta aos itens na habilidade Raciocínio Matemático em ambos algoritmos.	3476	2538	3330	2617	56	581	0,00384
Não existe diferença estatística no tempo de resposta aos itens na habilidade Flexibilidade Cognitiva em ambos algoritmos.	3494	1273	1622	1186	1262	1048	<0,00001

**Fonte:** Elaborada pelo autor (2022).

De acordo com os resultados da Tabela 7, respondemos **QP2**. O tempo de resposta aos itens usando o algoritmo adaptativo é menor quando comparado ao método tradicional? O gerenciamento adaptativo do banco de itens, utilizando o erro padrão de estimativa <0,01 e o número mínimo e máximo de itens como critérios de parada, reduziu consideravelmente o tempo de resposta para aplicação de itens para todas as habilidades investigadas, pois *p-value* é inferior a 0,05. Desta forma, este algoritmo oferecerá algumas vantagens para O Screening Programmin. Entre eles, a redução do número de itens administrados no instrumento que foi possível devido à administração correta dos itens e a precisão na estimativa da habilidade dos indivíduos.

## 5 CONSIDERAÇÕES FINAIS E SUGESTÕES PARA TRABALHOS FUTUROS

Este trabalho partiu da necessidade de integrar a seleção adaptativa em um instrumento para medir habilidades preditoras de programação, a fim de melhorar a eficiência do sistema na seleção de itens, bem como na estimativa de habilidades dos alunos. Para isso, desenvolvemos algoritmos de seleção adaptativa de itens, definindo um critério para a habilidade inicial, procedimento de seleção de itens, método de estimativa de habilidades e critérios de parada. Por fim, comparamos os algoritmos por meio de uma simulação utilizando o banco de dados do Screening Programming, a fim de verificar o desempenho em relação à estimativa de habilidade e ao tempo de resposta aos itens.

As principais questões de pesquisa deste estudo são: i) qual algoritmo adaptativo tem melhor precisão na estimativa de habilidades preditoras de programação? ii) o tempo de resposta aos itens usando o algoritmo adaptativo é menor quando comparado ao método tradicional?

Para entender melhor se é possível melhorar adaptativamente o processo de aplicação de um instrumento, selecionando itens de acordo com o nível de habilidade do sujeito, além de administrar menos itens do que o convencional, esta pesquisa obteve os seguintes resultados em cada questão específica dentro o escopo mais amplo:

- **(QP1.)** Entre os algoritmos desenvolvidos para selecionar itens de forma adaptativa, o algoritmo que obteve melhor precisão no processo de estimação das habilidades possui as seguintes características: i) máxima verossimilhança como método de estimação da habilidade; ii) máxima informação de fisher como método de seleção adaptativa; iii) obter um controle de exposição de itens; iv) e para o método de parada do instrumento, uma combinação de fatores como o erro padrão na estimação  $<0,01$  e a quantidade mínima e máxima de itens administrados;
- **(QP2.)** Com base nas características do algoritmo anterior, o tempo de resposta aos itens diminuiu consideravelmente para todas as habilidades investigadas.

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

- Obter outras variações do algoritmo com diferentes critérios para a seleção inicial dos itens, verificando, para tanto, qual obtém a melhor correlação com as estimativas de habilidades no pré-teste;
- Analisar a usabilidade do Screening Programming Adaptativo;
- Realizar um experimento a fim de verificar se, a partir da escala construída e do banco de itens, o sistema adaptativo provê o levantamento de características que possam identificar sujeitos com fraca/forte habilidade em tarefas preditoras de programação.

## 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 na página 16.
- 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 20.
- 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 20.
- ANDRADE, P. H. Análise da usabilidade de um sistema web na mensuração de habilidades cognitivas preditoras em novatos em programação. In: *Trabalho de Conclusão de Curso, Monografia (Bacharelado em Computação) – Centro de Ciências e Tecnologias, Universidade Estadual da Paraíba*. [S.l.: s.n.], 2022. Citado na página 13.
- 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 4 vezes nas páginas 18, 19, 20 e 21.
- 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 na página 17.
- BAKER, F. B. *The Basics of Item Response Theory*. [S.l.]: ERIC, 2001. Citado 3 vezes nas páginas 20, 21 e 25.
- BAKER, F. B.; KIM, S. *Item Response Theory: Parameter Estimation Techniques*. [S.l.]: CRC Press, 2004. Citado 2 vezes nas páginas 24 e 31.
- BAKER, F. B.; KIM, S.-H. *The basics of item response theory using R*. [S.l.]: Springer, 2017. Citado na página 19.
- 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 na página 16.
- 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 na página 16.
- COSTA, D. R. Métodos estatísticos em testes adaptativos informatizados. *Master's thesis, UFRJ-Universidade Federal do Rio de Janeiro*, v. 9, 2009. Citado na página 23.
- DANTAS, D. M. Análise da confiabilidade de um instrumento na mensuração de habilidades cognitivas preditoras em novatos em programação. In: *Trabalho de Conclusão de Curso, Monografia (Bacharelado em Computação) – Centro de Ciências e Tecnologias, Universidade Estadual da Paraíba*. [S.l.: s.n.], 2022. Citado 3 vezes nas páginas 13, 22 e 29.

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 na página 17.

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 na página 18.

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 na página 18.

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 na página 16.

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 na página 16.

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 na página 16.

HUTZ, C. S.; BANDEIRA, D. R.; TRENTINI, C. M. *Psicometria*. [S.l.]: Artmed Editora, 2015. Citado na página 19.

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 24.

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 24.

MLADENović, M.; KR PAN, D.; MLADENović, S. Learning programming from scratch. In: *Proceedings of the International Conference on New Horizons in Education*. [S.l.: s.n.], 2017. Citado na página 16.

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 na página 17.

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 na página 17.

PASQUALI, L. *Psicometria: Teoria dos Testes na Psicologia e na Educação*. [S.l.]: Editora Vozes Limitada, 2017. Citado 5 vezes nas páginas 18, 19, 21, 25 e 31.

PRIMI, R. *Psicometria: Fundamentos matemáticos da teoria clássica dos testes*. *Avaliação Psicológica*, Instituto Brasileiro de Avaliação Psicológica, v. 11, n. 2, 2012. Citado 2 vezes nas páginas 14 e 19.

SANDERS, K. et al. The canterbury questionbank: Building a repository of multiple-choice cs1 and cs2 questions. In: ACM CANTERBURY, UNITED KINGDOM. *Proceedings of the Innovation and Technology in Computer Science Education (ITiCSE)*. [S.l.], 2013. Citado na página 16.

SANDS, P. Addressing cognitive load in the computer science classroom. *ACM Inroads*, ACM New York, NY, United States, v. 10, n. 1, 2019. Citado na página 13.

SARTES, L. M. A.; SOUZA-FORMIGONI, M. L. O. de. Avanços na psicometria: da teoria clássica dos testes à teoria de resposta ao item. *Psicologia: Reflexão e Crítica*, Universidade Federal do Rio Grande do Sul, v. 26, n. 2, p. 241–250, 2013. Citado na página 14.

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 2 vezes nas páginas 14 e 18.

WAINER, H. et al. *Computerized adaptive testing: A primer*. [S.l.]: Routledge, 2000. Citado na página 14.

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 13.

## 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 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. 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)

## ANEXO C – SCRIPT

```

1  /*
2  Script para Selecao Adaptativa dos Itens e Estimativa da Habilidade
3  */
4  public class Script{
5      private static int iS, id, idStudent, idmirror, max = 10,
6          admint;
7
8      private static float difErro, erro, erro1, erro2, erroAux,
9          theta, numTheta, denTheta;
10
11     private static float[] a = new float[] {}; //Inserir os itens
12         calibrados para o parametro discriminacao
13
14     private static float[] b = new float[] {}; //Inserir os itens
15         calibrados para o parametro dificuldade
16
17     private static float[] c = new float[] {}; //Inserir os itens
18         calibrados para o parametro probabilidade de chute
19
20     //Probabilidade de Acertar ao Item
21     private static float[] p = new float[10];
22
23     //Funcao de Informacao dos Itens
24     private static float[] i = new float[10];
25
26     //Controle de Exposicao dos Itens
27     public static int[] ic = new int[] {0,0,0,0,0,0,0,0,0,0};
28
29     private static int[][] mirror = new int[][] {
30         {0,0,0,0,0,0,0,0,0,0},
31         {1,1,1,1,1,1,1,1,1,1},
32         //Inserir base de dados neste formato
33         {99,1,0,0,0,0,0,0,0,0}
34     };
35
36     public static void main(String[] args) {
37         iS = 0;
38         idStudent = mirror [iS] [0];
39         do{
40             admint = 0;
41             theta = 0.0f;

```

```

36         do{
37             newItem();
38             ic[id] = 1;
39             admint = admint + 1;
40             System.out.printf("Id: %d | Skill: %.3f | Item: %d
                               | DifErro: %.3f | Items: %d\n", idStudent, theta
                               , id, difErro, admint);
41         }while(admint<10); //--> Criterio de parada para Z2
42         //}while(admint<10 && difErro>0.01); --> Criterio de
           parada para Z3
43         //while(admint<5 || (admint<10 && difErro>0.01)); -->
           Criterio de parada para Z4
44
45         ic = new int[] {0,0,0,0,0,0,0,0,0,0};
46         iS = iS + 1;
47         if(iS<mirror.length){
48             idStudent = mirror [iS] [0];
49         }
50         denTheta = 0.0f;
51         numTheta = 0.0f;
52         erro = 0.0f;
53     } while(iS<mirror.length);
54 }
55
56 public static void newItem(){
57     probI();
58     infoI();
59     newTheta();
60 }
61
62 public static void probI(){
63     for (int index = 0; index < p.length; index++){
64         float den = (float) (1.0f + Math.exp(-a[index] * (theta
           - b[index])));
65         float pTheta = c[index] + ((1.0f - c[index]) / den);
66         p[index] = pTheta;
67     }
68 }
69
70 public static void infoI(){
71     float max = 0.0f;
72     for (int index = 0; index < i.length; index++){

```

```

73         float iItem = (a[index]*a[index])*((1.0f-p[index])/p[
              index])*((p[index]-c[index])*(p[index]-c[index]))
              /(((1.0f-c[index])*(1.0f-c[index])));
74         i[index] = iItem;
75     }
76
77     for (int index = 0; index < i.length; index++){
78         if (i[index] > max){
79             if (ic[index] == 0){
80                 max = i[index];
81                 id = index;
82                 idmirror = id + 1;
83             }
84         }
85     }
86 }
87
88 public static void newTheta(){
89     numTheta = numTheta + (a[id]*(mirror[iS][idmirror]-p[id]));
90     denTheta = denTheta + ((a[id]*a[id])*p[id]*(1-p[id]));
91     theta = theta + (numTheta/denTheta);
92     if (theta < -4.0f){
93         theta = -4.0f;
94     }
95     else if (theta > 4.0f){
96         theta = 4.0f;
97     }
98     erro = (float) (1/Math.sqrt(denTheta));
99
100    if (admint == 0){
101        erro1 = 0;
102        erro2 = erro;
103        difErro = erro2 - erro1;
104    } else if (admint > 0){
105        erro1 = erro;
106        erro2 = erroAux;
107        difErro = erro2 - erro1;
108    }
109    erroAux = erro;
110 }
111 }

```