



**UNIVERSIDADE FEDERAL DA BAHIA  
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA**

**DANILO SANTOS DOS SANTOS**

**O DESENVOLVIMENTO DE HABILIDADES DO RACIOCÍNIO  
COMPUTACIONAL NA APRENDIZAGEM DE PROGRAMAÇÃO POR  
ESTUDANTES INICIANTEs**

**SALVADOR – BAHIA**

**2018**

DANILO SANTOS DOS SANTOS

O DESENVOLVIMENTO DE HABILIDADES DO RACIOCÍNIO COMPUTACIONAL  
NA APRENDIZAGEM DE PROGRAMAÇÃO POR ESTUDANTES INICIANTE

Trabalho de Conclusão de Curso apresentado ao Colegiado do Curso de Graduação em Sistemas de Informação do Instituto de Matemática e Estatística da Universidade Federal da Bahia, como requisito parcial à obtenção do grau de bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Ecivaldo de Souza Matos

SALVADOR – BAHIA

---

Danilo Santos dos Santos

**O DESENVOLVIMENTO DE HABILIDADES DO  
RACIOCÍNIO COMPUTACIONAL NA  
APRENDIZAGEM DE PROGRAMAÇÃO POR  
ESTUDANTES INICIANTEs**

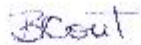
Este trabalho foi julgado adequado para a obtenção do grau de Bacharel no curso de Sistemas de Informação e aprovado em sua forma final pela Comissão examinadora no dia 30 de julho de 2018 na sala 106 da Superintendência Tecnologia da Informação desta Universidade e pelo Colegiado do Curso de Graduação em Sistemas de Informação da Universidade Federal da Bahia.

Comissão Examinadora.



Profa. Dra. Debra Abdala Santos | UFBA

Examinadora



Profa. Cláudia Borges Coutinho/SEC-BA

Examinadora



Prof. Ms. Renansamir Sousa da Silva | UFBA

Examinador

IN MEMORIAM

Minha madrinha Maria e minha vó Creuza.

## **AGRADECIMENTOS**

A esta universidade, seu corpo docente, direção e administração que oportunizaram a janela que hoje vislumbro um horizonte superior, eivado pela acendrada confiança no mérito e ética aqui presente.

Ao grupo Onda Digital e à Prof.<sup>a</sup>. Dra. Débora Abdalla pela oportunidade e apoio na execução deste trabalho.

Aos meus pais e irmão, pelo amor, incentivo e apoio incondicional.

Muito obrigado à Liliane Fiúza, Cíntia Mercês, Bárbara Fernandes e Luana Menezes pela ajuda nas correções ortográficas e paciência.

A todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado!

“Pensamento Computacional é uma habilidade fundamental para todos, não apenas para cientistas da Computação. Além de aprender a ler, escrever e calcular, deveríamos adicionar Pensamento Computacional na capacidade analítica de cada criança”

‘(Jeannette Wing)

## RESUMO

Como o Raciocínio Computacional pode ajudar a melhorar a aprendizagem de iniciantes em programação? Esta é uma pesquisa qualitativa que busca elucidar como as habilidades do Raciocínio Computacional alinhado as estruturas lógicas de programação contribuem na aprendizagem dos estudantes iniciantes. Aplicando-se intervenções referentes às habilidades do Raciocínio Computacional, em um grupo de estudantes de iniciação em programação, coletando dados através de questionários, entrevistas, grupo focal e avaliação, com objetivo de investigar como o desenvolvimento das habilidades alinhado ao ensino das estruturas lógicas de programação pode favorecer a aprendizagem dos estudantes. Buscou-se observar nos ingressantes do curso, quais as habilidades de Raciocínio Computacional possuíam carências e que ao decorrer das atividades as habilidades foram surgindo durante o andamento dos módulos. Ficou notório que a evolução decorrente das intervenções, facilitaram o aprendizado durante o curso, de forma que os alunos destacaram a importância das habilidades do Raciocínio Computacional também no seu dia-a-dia.

**Palavras-chave:** Raciocínio Computacional, Educação em Computação, Iniciantes em Programação

## **ABSTRACT**

How can Computational Reasoning Help Improve Beginner's Learning in Programming? This is a qualitative research that seeks to elucidate how the skills of Computational Reasoning aligned the logical structures of programming contribute in the learning of the beginner students. Applying interventions related to Computational Reasoning skills, in a group of students of initiation in programming, collecting data through questionnaires, interviews, focus group and evaluation, aiming to investigate how the development of skills aligned with the teaching of the logical structures of programming can favor student learning. It was sought to observe in the students of the course, which the abilities of Computational Reasoning had deficiencies and that during the course of the activities the skills arose during the progress of the modules. It was noted that the evolution of the interventions facilitated the learning during the course, so that the students emphasized the importance of the skills of Computational Reasoning in their daily lives.

**Keywords:** Computational Reasoning, Computer Education, Beginners in Programming



## LISTA DE FIGURAS

<b>Figura 1</b> – Pilares do Raciocínio Computacional .....	18
<b>Figura 2</b> – Modelos Abstração .....	19
<b>Figura 3</b> – Exemplo de carro decomposto .....	20
<b>Figura 4</b> – Código Morse.....	21
<b>Figura 5</b> – Modelo algoritmo de somar dois números.....	22
<b>Figura 6</b> – Percurso Metodológico .....	24
<b>Figura 7</b> – Motivo da escolha do curso de Programação.....	28
<b>Figura 8</b> – Possuía algum contato com programação antes desse curso. ....	28
<b>Figura 9</b> – Categorização das habilidades. ....	35
<b>Figura 10</b> – Comparação após a intervenção 01.....	39
<b>Figura 11</b> – Comparação após a intervenção 02.....	40

## LISTA DE TABELAS

<b>Tabela 1 – Habilidades .....</b>	<b>38</b>
-------------------------------------	-----------

## LISTA DE QUADROS

Quadro 1 – Calendário de aulas Turma A .....	26
Quadro 2 – Calendário de aulas Turma B .....	26
Quadro 3 – Características dos participantes do estudo .....	48
Quadro 4 – Características dos participantes do estudo .....	50

## SUMÁRIO

1	<b>Introdução</b> .....	14
1.1	Objetivos .....	15
2	<b>Fundamentação Teórica</b> .....	16
2.1	Raciocínio Computacional .....	16
2.2	Os quatro Pilares .....	17
2.2.1	<b>Abstração</b> .....	18
2.2.2	<b>Decomposição</b> .....	19
2.2.3	<b>Reconhecimento de Padrões</b> .....	20
2.2.4	<b>Algoritmo</b> .....	21
3	<b>Metodologia</b> .....	23
3.1	Revisão da Literatura .....	25
3.1.1	<b>Mapeamento Sistemático</b> .....	25
3.2	Caracterização do Ambiente .....	26
3.3	Caracterização dos sujeitos .....	27
3.3.1	<b>Instrutora da Turma A</b> .....	29
3.3.2	<b>Instrutora da Turma B</b> .....	29
3.4	Estratégia de ensino.....	29
3.5	Questionário: perfil do sujeito.....	29
3.6	Questionário: atividade de sondagem.....	30
3.7	Categorias de análise.....	30
3.7.1	<b>Categoria Suficiente</b> .....	31
3.7.2	<b>Categoria Parcial</b> .....	31
3.7.3	<b>Categoria Insuficiente</b> .....	31
3.8	Intervenções .....	31
3.8.1	<b>Primeira Intervenção</b> .....	32
3.8.2	<b>Segunda Intervenção</b> .....	32
3.8.3	<b>Atividades Intervenções</b> .....	32
3.9	Diário de Bordo.....	32
3.10	Entrevistas .....	33

3.11	Grupo Focal.....	33
3.12	Considerações Éticas.....	34
<b>4</b>	<b>Resultados e Discussões</b> .....	<b>35</b>
4.1	Questionário de Sondagem .....	35
4.2	Questionário de Diários de Bordo .....	36
4.3	Entrevistas .....	36
4.4	Grupo Focal.....	37
4.5	Intervenções .....	38
<b>4.5.1</b>	<b>Intervenção 01</b> .....	<b>38</b>
<b>4.5.2</b>	<b>Intervenção 02</b> .....	<b>40</b>
4.6	Resultados Individuais .....	40
<b>5</b>	<b>Conclusões e Trabalhos futuros</b> .....	<b>42</b>
5.1	Trabalhos Futuros.....	43
<b>6</b>	<b>Referências</b> .....	<b>44</b>
<b>APÊNDICES</b> .....		<b>46</b>
<b>APÊNDICE A</b> – Termo de Consentimento Livre e Esclarecido .....		<b>46</b>
<b>APÊNDICE B</b> – Questionário de Perfil.....		<b>47</b>
<b>APÊNDICE C</b> – Características dos Participantes .....		<b>48</b>
<b>APÊNDICE D</b> – Características dos Participantes Turma B .....		<b>50</b>
<b>APÊNDICE E</b> – Entrevista Docente 01.....		<b>51</b>
<b>APÊNDICE F</b> – Entrevista Docente 02.....		<b>55</b>
<b>APÊNDICE G</b> – Atividade Habilidades Abstração e Reconhecimento de Padrões.....		<b>59</b>
Perguntas do quiz.....		59
<b>APÊNDICE H</b> – Atividade Habilidades Decomposição e Algoritmos .....		<b>61</b>
Quiz Abstração e reconhecimento de padrões.....		61
Perguntas do quiz.....		61
<b>APÊNDICE I</b> – Grupo Focal .....		<b>64</b>
<b>APÊNDICE J</b> – Diário de Bordo.....		<b>71</b>

## **Introdução**

O primeiro contato com programação para muitos estudantes costuma ser frustrante. Dentre os motivos para esta frustração destacam-se: falta de uma visão do problema, preocupação excessiva com detalhes de sintaxe da linguagem utilizada; idealizar soluções adequadas, de mapear essas soluções em passos sequenciais e de abstrair o funcionamento dos mecanismos escolhidos; o estabelecimento de um raciocínio lógico visando à resolução de problemas, com base em um modelo incremental, em relação à complexidade e à estratégia de refinamentos sucessivos (PROULX, 2000).

A literatura na área de ensino e de aprendizagem em programação é unânime ao afirmar que começar a programar é considerado difícil pela maioria dos estudantes iniciantes nesta disciplina. Autores como Sheard *et al.* (2009) apontam a grande quantidade de literatura relacionada às dificuldades da programação introdutória como um indicador de que este assunto é difícil de aprender e de ensinar. As dificuldades tornam-se mais evidentes quando se observa taxas de reprovação entre 30% e 40% nestas disciplinas (BEAUBOUÉF; MASON, 2005).

Vários estudos vêm investigando as dificuldades na aprendizagem de programação introdutória sob diferentes aspectos (LAHTINEN *et al.* 2005). Contudo, o maior dos problemas vivenciados pelos iniciantes em programação não parece ser o entendimento dos conceitos básicos de lógica de programação, mas a combinação e a utilização adequada destes conceitos na construção de um determinado programa (LAHTINEN *et al.* 2005). Sendo assim, a maioria dos erros cometidos pelos estudantes não resulta de falhas no conhecimento sobre os construtos da lógica de programação, mas resulta de dificuldades em combiná-los corretamente, ou seja, “em colocar as peças do programa juntas” (SPOHRER; SOLOWAY, 1989, p. 401).

Programar é considerada uma tarefa tão complexa que estudos apontam que, mesmo após passar por uma disciplina introdutória de programação, estudantes ainda apresentam sérios problemas ao aplicar os conceitos da disciplina (KURLAND *et al.* 1986). Pesquisas também apontam que a maioria dos estudantes têm seus conhecimentos de programação consolidados apenas ao final da segunda disciplina de programação (TEW *et al.* 2005). Por esse motivo, ensinar programação é considerado um dos grandes desafios da área de educação em computação (CASPERSEN; KOLLING, 2009).

Aliado a isso, os estudantes de programação adotam metodologias de estudo que são ineficientes. Eles decoram soluções para os problemas ao invés de entendê-las (GOMES *et al.* 2012), estudam de maneira passiva ou superficial (BIGGS; TANG, 2011) e fazem poucos exercícios (GOMES *et al.* 2012). Além disso, os estudantes de programação possuem um tipo de conhecimento dito frágil, que é aquele conhecimento que eles possuem, mas não conseguem aplicar em novas situações ou problemas encontrados (PERKINS; MARTIN, 1986).

Diante dos dilemas relacionados, levantou-se o seguinte questionamento: Como o raciocínio computacional pode ajudar a melhorar a aprendizagem de iniciantes em programação?

### 1.1 Objetivos

Investigar a aprendizagem de iniciantes em programação a partir do desenvolvimento de habilidades do raciocínio computacional.

#### **Objetivos Específicos**

OE1. Investigar elementos compositores do raciocínio computacional.

OE2. Revisar elementos que caracterizam o interesse pela programação.

OE3. Analisar a influência do Raciocínio Computacional na aprendizagem de programação pelos iniciante

## 2. Fundamentação Teórica

Este capítulo destina-se a uma breve apresentação de conceitos fundamentais relacionados à Educação em Computação, com enfoque no Raciocínio Computacional e a apresentação de abordagens significativas relacionadas ao tema.

### 2.1 Raciocínio Computacional

No ano de 2006, Jeanette Wing popularizou o termo “*Computational Thinking*”<sup>1</sup> através do seu artigo publicado na Revista “*Communications*” (ACM) (WING, 2006). Nele, a autora expôs o que os Cientistas da Computação pensam sobre o mundo, os quais dizem que é útil para outros contextos. Ela não criou o termo, mas descreveu o que os Cientistas da Computação fazem, e o que a Ciência da Computação seria capaz de oferecer para as outras áreas leigas no tema.

Ribeiro *et al.* (2017) dizem que:

O grande objetivo da Computação é "raciocinar sobre o raciocínio". Porém, diferente da Filosofia, aqui não estamos pensando de forma mais ampla sobre o raciocínio, mas sim interessados no processo de racionalização do raciocínio, ou seja, formalização do mesmo, o que permite a sua automação e análise (matemática). Essa questão de racionalização do raciocínio está relacionada com a resolução de problemas (RIBEIRO *et al.* 2017, p. 1).

O Raciocínio Computacional pode ser facilmente confundido com o raciocínio lógico. Basicamente, o raciocínio lógico busca verdades para os problemas, partindo de axiomas bem definidos para encontrar novas conclusões. Por sua vez, o Raciocínio Computacional é a generalização do raciocínio lógico, ou seja, o Raciocínio Computacional busca transformar suas entradas em saídas, onde elas não necessariamente são sentenças verdadeiras, mas sim um conjunto de elementos de tipos iguais ou diferentes um do outro e através de uma sequência de regras que define a transformação, que geralmente chamamos de algoritmo, ou seja, o Raciocínio Computacional estaria relacionado ao raciocínio humano e a forma como ser humano processa informação. As regras aplicadas no Raciocínio Computacional não são necessariamente as regras da

---

<sup>1</sup>O termo pensamento computacional é uma tradução do termo original em inglês *computational thinking*. Embora do ponto de vista filosófico existam diferenças entre os termos raciocínio e pensamento, neste trabalho usaremos os dois termos como sinônimos.



lógica, mas um conjunto qualquer de regras ou instruções bem definidas, pois, o problema que está sendo esclarecido aqui é como transformar a entrada na saída.

Liukas (2015, p. 110) explica que “O Raciocínio Computacional é executado por pessoas e não por computadores”. Ele inclui ainda o pensamento lógico, a habilidade de reconhecimento de padrões, raciocinar através de algoritmos, decompor e abstrair um problema. A necessidade de definir o conceito de Raciocínio Computacional fez com que, a *International Society for Technology in Education* (ISTE) em conjunto com a *Computer Science Teachers Association* (CSTA) no ano de 2011 efetuassem a definição operacional de Raciocínio Computacional, após aprovação ficou definido da seguinte forma:

O Raciocínio Computacional é um processo de resolução de problemas que inclui (mas não está limitado a) as seguintes características:

- Formulação de problemas de forma que nos permita usar um computador e outras ferramentas para nos ajudar a resolvê-los;
- Organização e análise lógica de dados;
- Representação de dados através de abstrações, como modelos e simulações;
- Automatização de soluções através do pensamento algorítmico (uma série de etapas ordenadas);
- Identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a combinação mais eficiente e efetiva de etapas e recursos;
- Generalização e transferência deste processo de resolução de problemas para uma grande variedade de problemas.

Essas habilidades são apoiadas e reforçadas por uma série de qualidades ou atitudes que são dimensões essenciais do PC. Essas qualidades ou atitudes incluem:

- Confiança em lidar com a complexidade;
- Persistência ao trabalhar com problemas difíceis;
- Tolerância para ambiguidades;
- A capacidade de lidar com os problemas em aberto;
- A capacidade de se comunicar e trabalhar com outros para alcançar um objetivo ou solução em comum (ISTE/CSTA, 2011, p. 7).

A British Broadcasting Corporation (BBC) LEARNING (2018) resumiu os elementos que o Raciocínio Computacional tende a atender nos chamados “Quatro Pilares do Raciocínio Computacional”, sendo eles: Decomposição, Reconhecimento de Padrões, Abstração e Algoritmos. Cada um dos pilares será explanado na sequência.

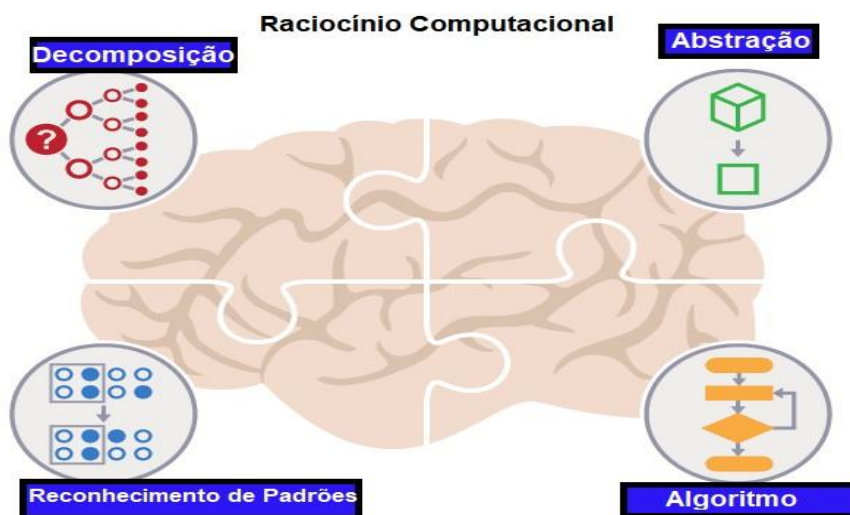
## 2.2 Os quatro Pilares

O Raciocínio Computacional envolve focar apenas nos detalhes importantes, enquanto informações irrelevantes são ignoradas (abstração), depois identifica-se um problema complexo e quebrá-lo em pedaços menores que são mais fáceis de gerenciar

(decomposição). Cada um desses problemas menores pode ser analisado individualmente com maior profundidade, identificando problemas parecidos que já foram solucionados anteriormente (reconhecimento de padrões). Por último, passos podem ser criados para resolver cada um dos subproblemas encontrados (algoritmos). Estas habilidades são igualmente importantes e dependentes entre si, entretanto, sem sequência definida durante a elaboração de soluções computáveis e viáveis.

Esses pilares, também conhecidos como habilidades do Raciocínio Computacional, pode ser observado na Figura 1.

**Figura 1**– Pilares do Raciocínio Computacional



Fonte: BBC LEARNING

### 2.2.1 Abstração

Esta habilidade engloba selecionar os dados e sua classificação, essencialmente omitindo elementos que não são necessários para focar nos que são importantes. Através desta técnica, pode-se criar uma representatividade do que está se buscando solucionar. A capacidade essencial deste pilar é selecionar a informação a ser ignorado para que o problema seja mais fácil de ser compreendido sem perder nenhuma informação que seja importante.

De acordo com Wing (2006), é o conceito mais importante do Raciocínio Computacional, pois, o processo de abstrair é utilizado em diversos momentos, tais como: na escrita do algoritmo e suas iterações; na seleção dos dados importantes; na escrita de uma pergunta; na alteridade de um indivíduo em relação a um robô; na compreensão e

organização de módulos em um sistema. Este pesquisador que aqui escreve compartilha da mesma ideia da autora.

A abstração proporciona um poder para escalonamento e tratamento com a complexidade (WING, 2006).

A Figura 2 mostra um exemplo de abstrações de informações e lei, por exemplo. Essa abstração ocorre de se selecionar o que não importa para ser ignorado. Criando a representação do que se procura resolver ou passar informação.

**Figura 2** – Modelos Abstração



Fonte:shutterstock.com

### 2.2.2 Decomposição

A decomposição é um processo pelo qual os problemas são decompostos em partes menores (LIUKAS, 2015). Trata-se de dividir algo complexo em pedaços menores, que são mais manejáveis e fáceis de entender. Essas partes menores podem, então, serem examinadas e resolvidas, uma vez que são mais fáceis de trabalhar.

Quando um problema não é decomposto, sua resolução se torna difícil no momento de lidar com várias fases diferentes ao mesmo tempo, torna-se bastante difícil sua gestão. Uma forma de simplificar a solução é segmentar em partes reduzidas e resolvendo-as, individualmente. Esta prática também eleva o foco aos detalhes.

Por exemplo, para corrigir um mal funcionamento de um carro, sua manutenção se torna mais fácil quando o mesmo é separado em pequenas peças.

A Figura 3 exibe o carro decomposto, o que é possível reconhecer de forma mais clara cada uma das suas partes e a funcionalidade delas.

**Figura 3** – Exemplo de carro decomposto

Fonte:shutterstock.com

Liukas (2015) declara que desenvolvedores utilizam frequentemente este método para dividir um algoritmo em trechos menores para simplificar sua compreensão e manutenção. Em um código-fonte, pode-se exemplificar a decomposição através de funções, procedimentos, objetos, módulos, entre outros. Esta técnica permite resolver problemas complexos de forma mais simples, favorece a compreensão de novas situações e proporciona lançar sistemas de grande porte.

### **2.2.3 Reconhecimento de Padrões**

O Reconhecimento de Padrões consiste em encontrar os paradigmas que existem no problema, isto é, características que alguns dos problemas dividem e que podem ser abordados para se alcançar a solução eficaz. A descrição de Reconhecimento de Padrões, segundo Liukas (2015), consiste em encontrar similaridades e padrões com o intuito de resolver problemas complexos de forma mais eficiente. Para isso, procura-se por elementos que sejam iguais ou muito similares em cada problema. Na literatura, o Reconhecimento de Padrões também pode estar associado ao termo “Generalização”.

Esta também é uma forma de resolução de problemas rapidamente, fazendo uso de soluções previamente definidas em outros problemas e com base em experiências passadas. Os questionamentos que devem ser feitos são: “Esse problema é parecido a um

outro problema que já tenha solucionado” ou “Como ele é diferente? ”. Estas perguntas são relevantes, visto que, pode se definir as estratégias como base nos padrões com outros problemas anteriores que já se foi solucionado. As soluções usadas em problemas anteriores podem ser adaptadas para solucionar uma variedade de problemas similares.

Um exemplo do reconhecimento de padrões a Figura 4, mostra como é o Código Morse. Ele possui um padrão de “batidas” e intervalo de tempo para que seja possível ao receptor decodificar a mensagem encaminhada.

**Figura 4 – Código Morse**

<b>A</b>	· -	<b>J</b>	· - - -	<b>S</b>	· · ·	<b>2</b>	· - - - -
<b>B</b>	- · · ·	<b>K</b>	- · -	<b>T</b>	-	<b>3</b>	· · · - -
<b>C</b>	- · · ·	<b>L</b>	· - · ·	<b>U</b>	· · -	<b>4</b>	· · · - -
<b>D</b>	- · ·	<b>M</b>	- -	<b>V</b>	· · · -	<b>5</b>	· · · · ·
<b>E</b>	·	<b>N</b>	- ·	<b>W</b>	· - -	<b>6</b>	- · · · ·
<b>F</b>	· · · ·	<b>O</b>	- - -	<b>X</b>	- · · -	<b>7</b>	- - · · ·
<b>G</b>	- - ·	<b>P</b>	· - - ·	<b>Y</b>	- · - -	<b>8</b>	- - - · ·
<b>H</b>	· · · ·	<b>Q</b>	- - - ·	<b>Z</b>	- · - ·	<b>9</b>	- - - - ·
<b>I</b>	· ·	<b>R</b>	· - ·	<b>1</b>	· - - - -	<b>0</b>	- - - - -

Fonte: pt-br.oskaras.wikia.com

Através do reconhecimento de padrões, é possível auxiliar a resolução de problemas e reproduzir esta resposta em cada um dos subproblemas, caso haja afinidade, quanto mais padrões, se consegue descobrir, mais eficiente e rápido a solução é encontrada.

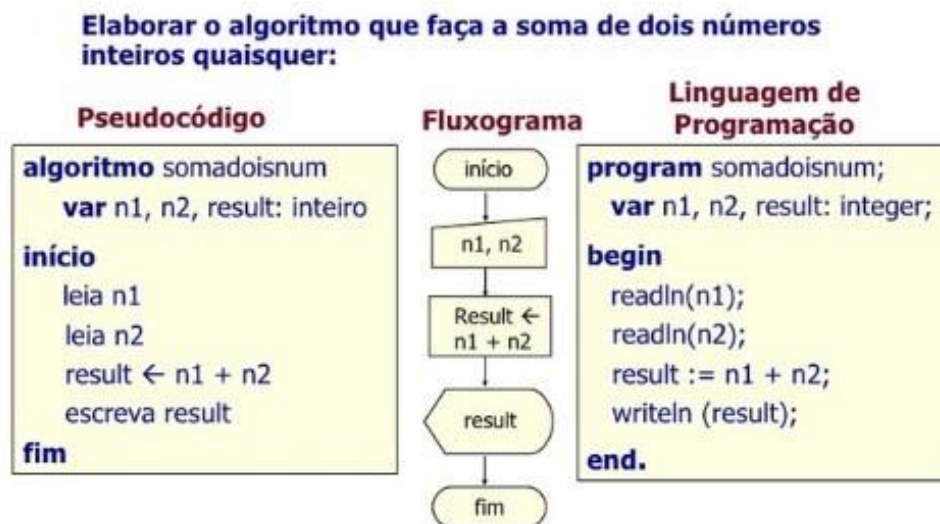
#### 2.2.4 Algoritmo

Apontado como o elemento que é agregador de todos os outros por Wing (2006), o algoritmo é um conjunto de instruções claras e bem definidas, fundamentais para a

solução de um problema. Em um algoritmo, as instruções são descritas e ordenadas para que o seu objetivo seja alcançado.

O algoritmo é um conjunto de regras para a resolução de um problema, como a receita de um bolo, a troca de uma lâmpada; porém, ao contrário de uma simples receita de bolo, pode-se utilizar diversos fatores mais complexos. Existem algoritmos muito pequenos, que podem ser comparados a pequenos poemas. Outros algoritmos são maiores e precisam ser escritos como se fossem livros, ou maiores ainda, necessitariam inevitavelmente serem escritos em diversos volumes de livros. A Figura 5 exemplifica um algoritmo de soma.

**Figura 5** – Modelo algoritmo de somar dois números.



Fonte: pt.slideshare.net

Algoritmos devem ser compreendidos como soluções prontas, pois, já passaram pelo processo de decomposição, abstração e reconhecimento de padrões para sua formulação. Ao serem executados, seguirão os passos pré-definidos, ou seja, aplicar-se-á solução quantas vezes forem necessárias, não havendo a necessidade de criar um novo algoritmo para cada uma de suas execuções posteriores.

### 3. Metodologia

Neste trabalho, caracterizou-se como um caso de estudo, um grupo de 15 alunos matriculados no curso de iniciação à programação de computadores do Programa Onda Digital<sup>1</sup>. Duas turmas foram estudadas, porque se acredita que a aprendizagem em programação é afetado pelo desenvolvimento de habilidades do raciocínio computacional. Para isso, foi analisado como o raciocínio computacional pode ajudar a melhorar a aprendizagem dos iniciantes em programação. Não foi necessário o estudo de muitos casos, mas aprofundamentos em alguns casos. Os casos serão tratados como um único grupo.

Creswell (2013) menciona que a coleta de dados é um processo amplo, que envolve inúmeros passos, como é o caso de ganhar permissões de acesso aos locais de coleta, definir as estratégias para selecionar os participantes, planejar os meios para gravar e manter as informações e antecipar questões éticas que podem se estabelecer ao longo do projeto.

A coleta de dados ocorreu através de um desenho longitudinal. Com os participantes divididos em duas turmas, foi utilizada uma abordagem combinada para o processo de mensuração da aprendizagem, que consistia em perfil dos sujeitos, atividades de sondagem, intervenções, diários de bordo, grupo focal e entrevistas com os instrutores, estes foram os instrumentos centrais do processo de coleta. Cada instrumento contribuiu de forma complementar para criar um senso sobre os dados.

Nesta pesquisa, os casos diferem entre si em muitas circunstâncias, mas possuem características em comum: são ofertados pelo mesmo programa e na mesma instituição federal de ensino e, por estarem localizados no mesmo centro, a infraestrutura de laboratórios, cronogramas e ambiente de aprendizagem é o mesmo.

Os casos foram estudados por três meses (período do curso). Foi considerada como uma unidade de análise um estudante do curso de iniciação à programação de computadores. Os participantes foram recrutados no Programa Onda Digital, na Universidade Federal da Bahia (UFBA), *Campus* de Ondina.

---

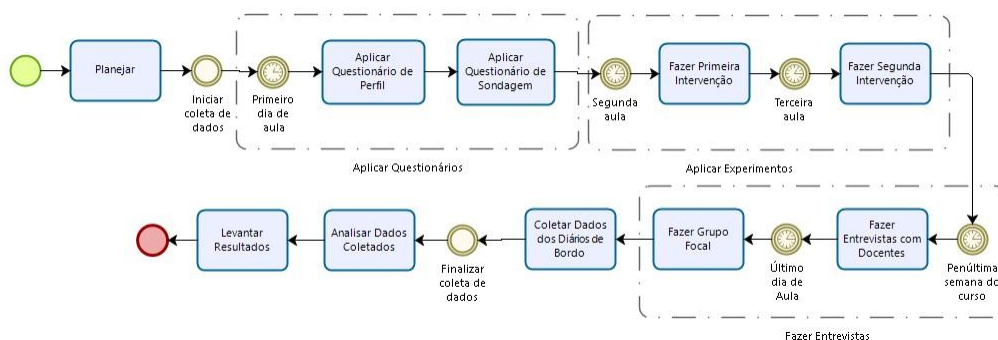
<sup>1</sup> O Programa Onda Digital (POD) foi criado em 2004, no âmbito do Departamento de Ciência da Computação (DCC) do Instituto de Matemática e Estatística, como um programa permanente de extensão da Universidade Federal da Bahia (UFBA).

Antes de iniciar o processo de coleta dos dados, houve uma explicação a todos que se tratava de uma pesquisa de cunho acadêmico e que eles poderiam desistir da participação em qualquer momento do processo de pesquisa. Os alunos assinaram um termo de consentimento (vide Apêndice A) que confirmava o interesse em participar voluntariamente da pesquisa.

O primeiro questionário entregue foi de atividade de sondagem. Nele continha seis questões (duas de múltipla escolha e quatro discursivas) que procurava saber informações sobre quais habilidades do Raciocínio Computacional o participante já possuía. Em seguida, foi aplicado o Questionário sobre as habilidades do raciocínio computacional na melhoria do aprendizado de iniciantes em programação, este procurava saber informações sobre o perfil dos usuários. Ambos os questionários foram aplicados em maio de 2018.

A condução da pesquisa foi estruturada conforme Figura 6. A primeira parte refere-se ao planejar desta pesquisa, com objetivo de efetuar uma revisão da literatura. A segunda parte refere-se a aplicar questionários de Perfil e questionário de Sondagem, com objetivo de identificar elementos que caracterizam o interesse pela programação. A terceira parte refere-se a aplicar experimentos. A quarta parte refere-se a fazer entrevistas, a terceira e quarta partes por sua vez, foram responsáveis por analisar a influência do Raciocínio Computacional na aprendizagem de programação pelos iniciantes em programação.

**Figura 6 – Percorso Metodológico**



Fonte: Próprio autor.

Nas seções apresentadas a seguir estão descritas com detalhes cada uma das fases, etapas e sub-etapas que compõe o processo de condução da pesquisa ilustradas na Figura 6.



### 3.1 Revisão da Literatura

Segundo Gil (2002) a pesquisa de caráter bibliográfico prospera apoiada em materiais existentes e são construídos com base em artigos científicos e livros. Nesse sentido, como primeira fase deste trabalho, foi desenvolvida uma pesquisa bibliográfica com o objetivo de buscar identificar na literatura as seguintes linhas de questionamentos: o que é Raciocínio Computacional e a diferença entre raciocínio lógico e raciocínio computacional.

Ainda na revisão da literatura buscou-se responder as seguintes perguntas: quais os pilares do Raciocínio Computacional? Qual é a diferença entre lógica e Raciocínio Computacional?

#### 3.1.1 Mapeamento Sistemático

Um mapeamento sistemático busca mostrar a quantidade ou frequência de publicações dentro de um determinado esquema. Dessa forma, a cobertura de uma área de pesquisa pôde ser determinada.

Kitchenham (2004) explica que o mapeamento sistemático é uma forma de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes para uma questão de pesquisa particular. Uma das razões para a realização de revisões sistemáticas é que esta resume as evidências existentes em relação a um tratamento ou tecnologia. Sendo assim, foram desenvolvidas as seguintes estratégias:

1. Bases científicas relacionadas ao tema: raciocínio computacional:
  - a. Periódicos CAPES/MEC;
  - b. Scielo (Scientific Electronic Library Online);
  - c. Google Acadêmico.
2. O idioma escolhido:
  - a. Português;
  - b. Inglês.

O autor Kitchenham (2004) diz que precisa adotar critérios de inclusão e exclusão para os artigos. Como base nisso foi definido também as seguintes regras de inclusões e exclusões:

1. Inclusão de Artigos

- a. Artigos escrito nos idiomas escolhidos;
- b. Artigos oportunos com a finalidade.

## 2. Exclusão de Artigos

- a. Artigos duplicados;
- b. Artigos indisponíveis ou inacessíveis;
- c. Leitura cinza.

### 3.2 Caracterização do Ambiente

O estudo de caso ocorreu em duas turmas do Curso de Extensão de iniciação à programação de computadores do Programa Onda Digital da UFBA (Universidade Federal da Bahia). Tal curso foi ofertado para pessoas com idade mínima de 15 anos, com conhecimentos em informática básica e que não necessitavam de conhecimentos em programação. Estas aulas ocorreram em um espaço de aula do programa Onda Digital, o qual possuía sete computadores com software livre instalado e acesso à internet. Os alunos não possuíam apostila, mas ao final das aulas os slides eram disponibilizados no ambiente virtual ao final.

As aulas da Turma A foram realizadas aos sábados com duração de três horas e eram aplicadas no horário das 13:30 às 16:30. A Turma B as aulas possuíam duração de duas horas e ocorriam nas terças e quintas-feiras no horário das 18:30 às 20:30. As turmas A e B foram regidas por dois instrutores diferentes, mas possuíam a mesma ementa em momento oportuno, sendo as aulas disponibilizadas nas seguintes datas:

**Quadro 1** – Calendário de Aulas Turma A

05/05/2018	12/05/2018	19/05/2018	26/05/2018	02/06/2018
09/06/2018	16/06/2018	30/06/2018	07/07/2018	14/07/2018

Fonte: <http://wiki.dcc.ufba.br/OndaDigital/Edital022018>

**Quadro 2** – Calendário de Aulas Turma B

08/05/2018	10/05/2018	15/05/2018	17/05/2018	22/05/2018
24/05/2018	29/05/2018	05/06/2018	07/06/2018	12/06/2018
14/06/2018	19/06/2018	21/06/2018	03/07/2018	05/07/2018

Fonte: <http://wiki.dcc.ufba.br/OndaDigital/Edital022018>

As turmas A e B foram regidas por dois instrutores diferentes, mas possuíam a mesma ementa descrita a seguir:

### **Ementa do Curso de Iniciação à Programação de Computadores (NÍVEL 1)**

1. Raciocínio lógico.
2. Algoritmos.
3. Expressões aritméticas.
4. Desvios condicionais.
5. Laços de repetição.
6. Vetores.
7. Ferramentas computacionais voltadas para o ensino de programação.
8. Implementação de soluções a problemas utilizando linguagem de programação.

#### 3.3 Caracterização dos sujeitos

Como já mencionado anteriormente, esta pesquisa contemplou um estudo de caso, contendo quinze unidades de análise. Referente à idade dos voluntários foi constatado que: cinco estão na faixa dos 15 aos 20 anos, quatro estão na faixa dos 21 aos 26 anos, cinco estão na faixa dos 27 aos 40 anos e um acima de 41 anos.

Portanto, para mapear o perfil destes estudantes foi aplicado um questionário solicitando alguns dados do entrevistado (vide Apêndice B). O questionário de perfil buscava entender as motivações que levaram os estudantes a escolherem o curso de iniciação à programação.

Essas informações facilitam a interpretação do comportamento dos discentes ao longo do curso e podem ser vista na Figura 7.

A Figura 7 apresenta dados sobre os motivos da escolhas dos discentes. A maioria dos participantes (10 pessoas) afirmaram que o interesse na área os motivaram a procurar o curso.

A Figura 8 mostra que 67% não tiveram contato prévio com programação, enquanto que 13% tiveram contato através de conteúdo na internet, outros 13% tiveram contato através de curso ou faculdade e 7% informaram que por necessidade de trabalho. No Apêndice C e Apêndice D contém algumas características dos participantes. Estes

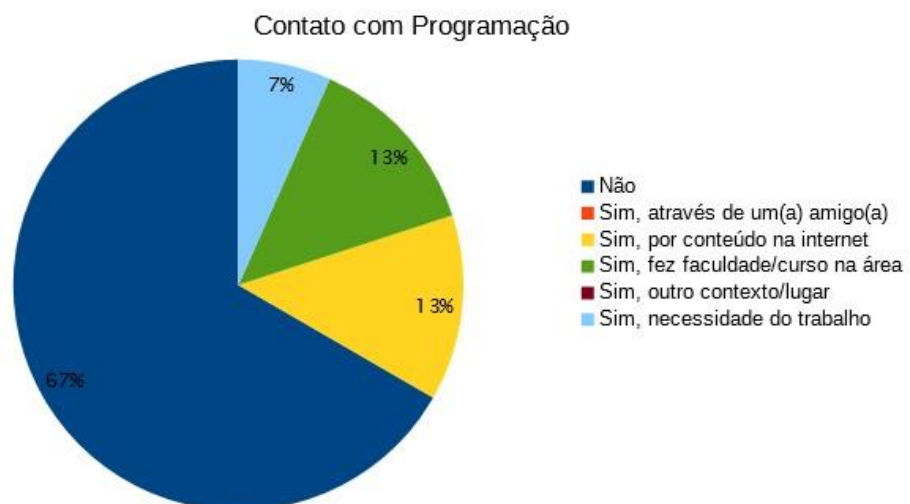
voluntários foram divididos em duas turmas para efeito de pesquisa, classificados como Turma A e Turma B.

**Figura 7** – Motivo da escolha do curso de Programação.



Fonte: Próprio autor.

**Figura 8** – Possuía algum contato com programação antes desse curso.



Fonte: Próprio autor.

### 3.3.1 Instrutora da Turma A

A instrutora é uma estudante de Bacharelado Interdisciplinar de Ciências e Tecnologia, encontra-se no seu terceiro ano de vivência em sala de aula e seu primeiro ano à frente de uma turma como responsável. Nos anos anteriores, foi monitora auxiliando o instrutor nas aulas de exercícios e no outro ano dividiu uma turma com outra instrutora. Segundo relatos dela, a experiência de estar à frente desta turma foi considerado ímpar para a instrutora que busca futuramente uma segunda graduação na área de licenciatura em computação (vide Apêndice E).

### 3.3.2 Instrutora da Turma B

Na Turma B estavam cursando sete participantes e sua instrutora é uma estudante de Bacharelado em Ciências da Computação. Tem experiência como instrutora de três anos. Com o foco em programação, esta é a primeira turma que leciona, mas foi monitora de outra turma de iniciação a programação (vide Apêndice F).

## 3.4 Estratégia de ensino

O método de ensino dos dois grupos se baseia no uso de slides para a exploração teórica e listas de exercício. As aulas são ministradas sempre em laboratório. Também são passadas atividades para o aluno fazer em casa, estas atividades são distribuídas em um ambiente virtual de ensino que possui uma estrutura de gamificação. Ao final do curso a Turma A efetuou um Dojo como trabalho final, onde a turma foi dividida em pares e respondiam a questões em um determinado tempo. Na Turma B o trabalho final foi tema livre, onde o aluno escolheria um tema e aplicaria todos os seus conhecimentos obtidos durante o curso para elaboração de um *quiz* do tema.

## 3.5 Questionário: perfil do sujeito

Foi solicitado aos participantes no mês de maio de 2018, que preenchessem um questionário de perfil (vide Apêndice B), contendo perguntas objetivas como por

exemplo: O que lhe motivou a escolher esse curso de Programação? ”; “Você já teve algum contato com programação antes desse curso?”. Além de perguntas abertas como: “Qual é a sua idade? ” ou “Qual é o seu nome completo?”.

O objetivo foi estabelecer um panorama inicial sobre cada participante e também o coletivo dos grupos. O questionário de perfil esteve disponível na plataforma Google Forms e ficou disponível nos primeiros dez minutos iniciais da primeira aula de cada turma. Neste questionário levantou-se questões sociais do indivíduo como idade, o que o motivou a se matricular no curso e se possuía contato com programação.

### 3.6 Questionário: atividade de sondagem

Nesta etapa, foi aplicado um questionário de sondagem de Raciocínio Computacional, simultaneamente a todos os alunos, sem interferência do pesquisador, com duração média de 50 minutos. Com objetivo de obter o conhecimento prévio dos sujeitos em relação as habilidades de Raciocínio Computacional no primeiro dia do curso. Estas informações foram classificadas em categorias de conhecimento para que fosse possível acompanhar a desenvoltura dos participantes durante o curso. O questionário ficou disponível no Google Forms apenas no período da aula (vide Apêndice B).

### 3.7 Categorias de análise

De acordo com Bardin (1995, p. 42), a análise de conteúdo é: “um conjunto de técnicas de análise das comunicações que visa obter, por procedimentos sistemáticos e objetivos de descrição do conteúdo das mensagens, indicadores que permitam a inferência de conhecimentos relativos às condições de produção destas mensagens”.

Para que fosse possível a análise, adotou-se a Análise Temática (AT) como método para identificar, analisar e reportar os padrões encontrados nos dados. Segundo Bardin (1995), a análise temática é uma das formas que melhor se adequou a investigações qualitativas. Na sequência, serão explicitadas as categorias de análise definidas para a realização da análise das atividades aplicadas ao conjunto de alunos.

### **3.7.1 Categoria Suficiente**

Buscou-se categorizar neste tema todos aqueles que utilizaram de maneira satisfatória as habilidades de Raciocínio Computacional, direta ou indiretamente para a solução(ões) do(s) problema(s) proposto(s). Entendeu-se como Suficiente todos aqueles que desenvolveram as respostas certas e explicou seu raciocínio de maneira clara e logica.

### **3.7.2 Categoria Parcial**

Foram classificado nesta categoria aqueles que realizaram parte das habilidades de Raciocínio Computacional esperado, direta ou indiretamente para a solução (ões) do(s) problema(s) proposto(s). Entendeu-se como Parcial todos aqueles que desenvolveram as respostas certas, mas não foram claros na explicação de seu raciocínio e logica.

### **3.7.3 Categoria Insuficiente**

Neste tema categorizou-se todos os que apresentaram escassez das habilidades de Raciocínio Computacional esperado, direta ou indiretamente para a solução(ões) do(s) problema(s) proposto(s). Entendeu-se como Insuficiente todos aqueles que não desenvolveram as respostas certas, não explicaram seu raciocínio e logica ou explicaram de maneira errada para questão solicitada.

## **3.8 Intervenções**

As intervenções ocorreram em maio de 2018 e foram divididas em duas etapas. Cada etapa constituiu-se de uma aula referente à duas habilidades do raciocínio computacional e uma atividade no final que buscava coletar dados a respeito do conhecimento adquirido do assunto pelos alunos.

Na primeira etapa aplicou-se as habilidades de Abstração e Reconhecimento de Padrões e na segunda etapa as habilidades de Decomposição/Composição e Algoritmos. Em ambas as intervenções o tempo limite estabelecido foi de uma hora para cada etapa. Essas intervenções tinham como objetivo responder as questões levantadas por esta pesquisa.

### **3.8.1 Primeira Intervenção**

A primeira intervenção ocorreu no segundo dia de aula, como citado anteriormente, essa intervenção foi referente às habilidades de Abstração e Reconhecimento de Padrões.

Foi desenvolvida nos primeiros cinquenta minutos da aula uma explanação referente à Abstração e Reconhecimento de Padrões com teoria e exemplos respondidos em conjunto com os alunos, logo após foi disponibilizada uma atividade que ficou disponível apenas no período da aula para que os discentes pudessem responder, na qual continha questões de Abstrações e Reconhecimento de Padrões. Foram disponibilizados dez minutos para resposta da atividade.

### **3.8.2 Segunda Intervenção**

A segunda intervenção ocorreu no terceiro dia de aula, a qual foi desenvolvida na primeira hora de aula e foi dividida da seguinte maneira: cinquenta minutos para teoria, onde eram explicadas as habilidades e sua importância na programação e exemplos de uso delas na programação e no dia-a-dia. Logo após, foram dados dez minutos para responder uma atividade disponível no Google Forms que ficou disponível apenas no período da intervenção.

### **3.8.3 Atividades Intervenções**

As duas intervenções possuíam ao final uma atividade que era composta de questões objetivas e questões abertas. Essas atividades tinham como objetivo identificar o que os alunos tinham acabado de absorver das habilidades ensinadas. As atividades estão disponíveis nos Apêndices G (Atividade da primeira intervenção) e Apêndices H (Atividade da segunda intervenção).

## **3.9 Diário de Bordo**

Para acompanhar a desenvoltura dos alunos no curso, fez-se uso de diários de bordo, que são relatos das aulas, fornecidos pelos instrutores das turmas ao final de cada



aula, através da plataforma Moodle do programa Onda Digital, onde foi observado a evolução dos participantes. O uso de diários de bordo foi fundamental na coleta e na obtenção de dados a respeito do desempenho dos alunos durante o curso. Teve como objetivo compreender sobre a visão das instrutoras a desenvoltura dos alunos nas aulas ao longo do curso.

Cabe salientar que, durante o desenvolvimento do curso, houve algumas greves de rodoviários e caminhoneiros, o que acarretou em cancelamento de aulas em alguns dias e conseqüentemente não existindo diário de bordo nesses dias.

### 3.10 Entrevistas

As entrevistas foram feitas com os instrutores das Turmas A e B ao final do curso, para colher “*feedbacks*” do desenvolvimento da turma e obter uma resposta sobre o quanto a pesquisa ajudou no decorrer do curso e no desenvolvimento, quanto ao aprendizado, dos alunos.

Teve também como objetivo trazer à tona circunstâncias que poderiam ter sido influentes no decorrer do curso, além de dicas para melhorar as intervenções. Para efeito de análise, as entrevistas foram gravadas com o consentimento dos instrutores e foram feitas uma série de perguntas levando em consideração tudo que foi ensinado aos alunos.

### 3.11 Grupo Focal

Um grupo focal com os participantes foi realizado com objetivo de coletar as experiências destes durante o período do curso.

Ao final do curso foi desenvolvida uma entrevista focal com objetivo de saber dos participantes o quanto o Raciocínio Computacional ajudou no projeto final do curso, bem como suas dificuldades.

Na Turma A houve o convite para realização do grupo focal, entretanto os participantes informaram estarem cansados da realização do Dojo e de outras pesquisas que aconteceram no dia e solicitaram a não participação.

Entretanto, através de relatos informais dos alunos foi constatada a importância das intervenções durante o curso. Reforçando o relato que a instrutora da turma havia informado em entrevista anterior (vide Apêndice E).

Na Turma B foi realizado o grupo focal com os concluintes do curso que totalizaram três alunos (vide Apêndice I). Para efeito de análise, as entrevistas focais foram gravadas.

### 3.12 Considerações Éticas

Por envolver seres humanos, a pesquisa seguiu os procedimentos normatizados pela resolução 196/96 do Conselho Nacional de Saúde. Antes de iniciar a coleta de dados, cada participante leu e assinou o Termo de Consentimento Livre e Esclarecido, que pode ser consultado no (vide Apêndice A).

Com o intuito de preservar critérios éticos, todos dados foram tornados anônimos. Como será observado, o texto desta pesquisa foi escrito de maneira a inviabilizar a identificação dos participantes e, portanto, a quebra da confidencialidade das informações. Desta forma, os nomes utilizados na narrativa são fictícios. Ao longo do texto, são utilizados pronomes no masculino ou feminino para se referir aos instrutores das turmas do CIPROG, mas não é verdadeiro assumir que este é o gênero que eles declararam possuir.

## 4. Resultados e Discussões

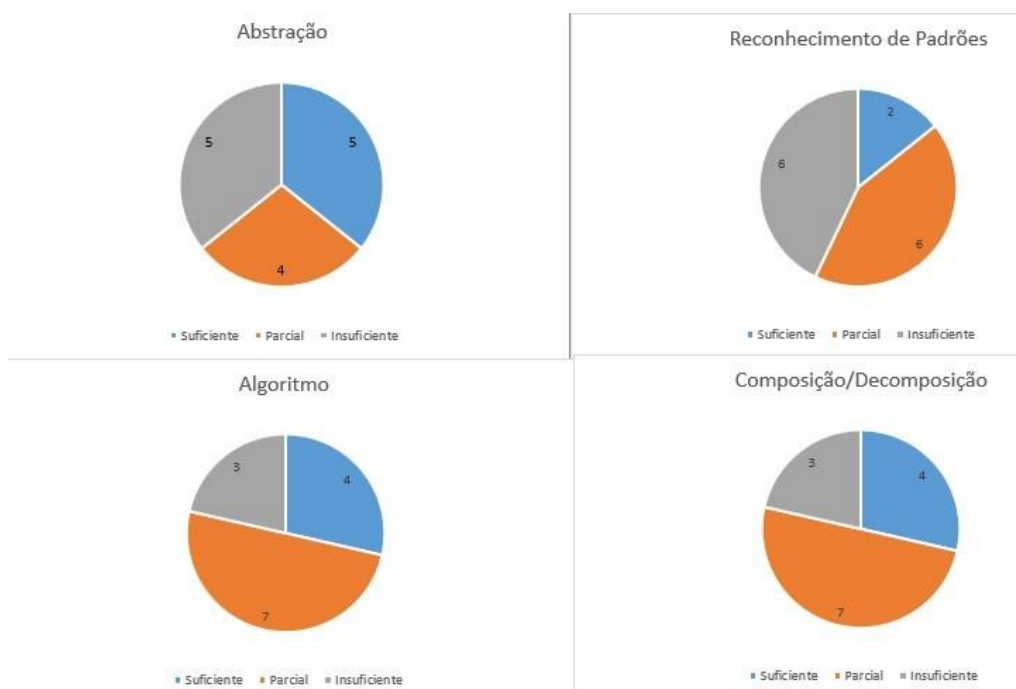
Esta pesquisa iniciou-se com 14 voluntários e foi concluída com 09 participantes. Até a conclusão deste trabalho alguns alunos estavam pendentes de entregas do seu trabalho final. Os resultados obtidos serão debatidos por ferramentas de coletas a seguir.

### 4.1 Questionário de Sondagem

Por intermédio do Questionário de Sondagem foi possível analisar o conhecimento prévio dos sujeitos a respeito das quatro habilidades propostas pelo Raciocínio Computacional.

A Figura 9 mostra dentro do estudo de caso a habilidade de Abstração, onde cinco foram categorizados como insuficiente, 04 parciais e cinco suficientes; já em Reconhecimento de Padrões foram 02 suficientes, 06 parciais e 06 insuficientes; no Algoritmo ficou distribuído por 04 suficientes, 07 parciais e 03 insuficientes; dentro da Composição e Decomposição ficou repartido com 04 participantes categorizados como suficientes, 07 parciais e 03 insuficientes.

**Figura 9** – Categorização das habilidades.



Fonte: Próprio autor.

#### 4.2 Questionário de Diários de Bordo

Após a análise dos diários de bordo das turmas (vide Apêndice J), podem ser observados os relatos dos professores e de suas equipes sobre o desenvolvimento dos alunos e a influência das intervenções aplicadas conforme transcrições<sup>1</sup> na sequência.

Na aula do dia 05 de Maio, tem-se o seguinte relato:

*“...A princípio o Danilo, fez uma intervenção sobre raciocínio computacional com elas, sem que antes fossem apresentados conceitos básicos quem são elencados logo na primeira aula. Seguindo a aula elas interagiram lembrando de conceitos e pontos apresentados pelo Danilo sobre raciocínio computacional e relacionaram com os conceitos de lógica de programação. Finalizamos com a explicação e inscrição das meninas no moodle da turma.” (Turma B)*

Na aula do dia 15 de maio, apresentou-se o seguinte registro:

*“Os alunos atenderam a intervenção de Danilo, fizeram a atividade sem dificuldade, e compreenderam o assunto.” (Turma A)*

Na aula do dia 09 de junho, pôde-se destacar a seguinte fala:

*“...A intervenção feita por Danilo nos ajudou bastante, pois fez com que as meninas relembassem de alguns conceitos que já haviam sido apresentados...” (Turma B)*

Já na aula do dia 16 de junho, a evolução foi comentada da seguinte maneira:

*“...Os exercícios aplicados em sala tem feito as meninas evoluir com mais rapidez no conteúdo, e gerando dúvidas pertinentes que nós conseguimos sanar na hora. Nossas alunas participam bastante da aula, em especial Aurora, essa aluna é a que mais questiona em sala e sempre tem curiosidades que vão além do conteúdo.” (Turma B)*

#### 4.3 Entrevistas

As entrevistas se mostraram fonte rica em dados desta pesquisa. Nelas foram constatados o quanto as intervenções ajudaram os professores nas aulas e foi possível colher informalmente alguns relatos de alunos para seus docentes. Todas as entrevistas podem ser vistas nos Apêndices E e F:

*“... Olha, eu achei que foi algo que realmente agregou porque durante muitas aulas após ter feito intervenções, elas usaram exemplos das*

---

<sup>1</sup> Em todas as transcrições e/ou excertos de fala são utilizados nomes fictícios.

*coisas que tinha acabado de falar durante a aula. Por exemplo: Fica o entendimento de algum conteúdo...” (Docente 01)*

*“É essencial, como falei antes, elas mostram dificuldades com coisas básicas de lógica, então quando você consegue trabalhar com o aluno questão de lógica eles não levam somente pra programação eles conseguem aplicar em outros setores e eles pensam nessa estrutura até mesmo quando eles partem pra outras matérias. Ex: Tinha duas alunas do ensino médio, se eu não me engano, e isso pode fazer um grande diferencial pra elas.” (Docente 01)*

*“Eu acho que quando você conseguiu suas intervenções com ela sem que os conceitos fossem apresentados de um modo tão formal. Isso conseguiu fazer o diferencial para o entendimento delas, tanto é, que elas conseguiram levar pra aulas e eu creio que realmente ajudou e mudou a visão que elas tinham do curso que só entende o que é fazer um programa de só entender o que é sentar e passar algo pra C.” (Docente 01)*

*“Logo nos assuntos iniciais de if. Onde você conseguiu fazer a intervenção sobre abstração e elas realmente pensaram nisso no momento de olhar pra o problema e tentar solucionar, foi uma coisa que marcou porque elas realmente usaram o termo abstrair pra solucionar o problema.” (Docente 01)*

*“Sim, porque com o ensino computacional elas conseguiriam desenvolver melhor as concepções delas lógicas e é uma coisa que como elas não conseguiram progredir bem no início do curso até porque tivemos essas pausas, agora elas apresentaram um pouco de dificuldade coma sequencia pra resolver um problema, entender a parte do problema, elas sabem até fazer mas não sabem como estruturar aquilo logicamente. ” (Docente 01)*

#### 4.4 Grupo Focal

O resultado alcançado com o Grupo Focal mostrou que os estudantes efetuavam o Raciocínio Computacional, porém inconscientemente (vide Apêndice I).

*“Eu achei assim pra nós, pra nós não, pra mim, vou falar por mim, pra nós que não viemos desse lugar, da ciência da computação, da programação, ela nos ajuda a se familiarizar com alguns conceitos, alguns termos usados e que dizem respeito a prática em si (...). Muitas dessas intervenções terem assim (...) serem teóricas né? Abstração da prática ela acaba nos ajudando num pouco do que ela falou assim, no sentido de nomear coisas que a gente faz, que agente as vezes não entende bem né?” (Gabriel)*

*“É, mas ajuda a pensar em estratégia mesmo, assim, por exemplo, a questão do perceber os padrões, né? Com isso, é super prático quando a gente vai aplicar, a gente entende por exemplo, quando a gente faz um quiz, e ontem a gente fez um exercício e o outro a gente sabe que existe um padrão, ainda que algumas coisas mudem, coisas se repetem*

*e aí aquilo que aparece na teoria e na prática fica muito evidente, quando a gente coloca a mão na massa mesmo, a mão no código.”*  
(Igor)

#### 4.5 Intervenções

Sobre esta etapa, foi coletado que do grupo inicial cinco participantes informaram já terem contato com programação e nove não tinham contato com programação antes. Destacou-se dos cinco participantes que informaram possuir contato com programação a classificação de insuficiente para Abstração e Reconhecimento de Padrões e Suficiente apenas para Algoritmo.

Já dos outros 09 participantes notou-se que a classificação parcial foi maioria em três habilidades e na habilidade de Abstração foi categorizado como suficiente.

Desta análise prévia concluiu que, quando se tratou da habilidade de decomposição e composição, 50% foram agrupados como parcial. Em contrapeso, 43% possuíam categorização da habilidade de abstração como insuficientes. Esses dados podem ser visto abaixo na Tabela 1:

**Tabela 1** – Habilidades

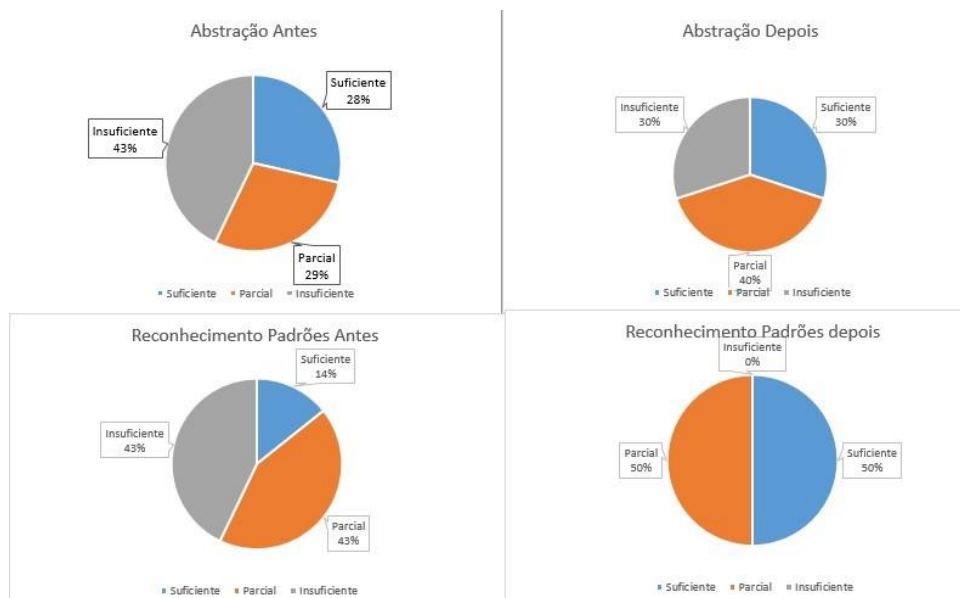
Habilidades	Suficiente	Parcial	Insuficiente
Abstração	4	4	6
Algoritmo	6	7	1
Composição/Decomposição	4	7	3
Reconhecimento de Padrões	2	6	6

Fonte: Próprio autor.

##### 4.5.1 Intervenção 01

Após a primeira intervenção, o estudo de caso possuía apenas 10 alunos ativos na pesquisa, os quais pôde-se observar uma evolução visível no grupo quanto as habilidades propostas pela intervenção como mostra a Figura 10 abaixo:

**Figura 10** – Comparação após a intervenção 01.



Fonte: Próprio autor.

Proporcionalmente, o número de participantes classificados como suficientes se manteve o mesmo, porém houve uma redução nos categorizados como insuficientes de 13%. Na competência de Reconhecimento de Padrões, o número dos capacitados como tal cresceu proporcionalmente 3 vezes em comparação ao início do curso e o número dos insuficiente caiu para zero.

A evolução não foi apenas momentânea, mas algo que foi visível e identificado por docentes e alunos durante o curso:

*"A intervenção feita por Danilo nos ajudou bastante, pois fez com que as meninas relembassem de alguns conceitos que já haviam sido apresentados". (Turma B)*

*"Logo nos assuntos iniciais de if/else, onde você conseguiu fazer uma intervenção sobre Abstração e elas realmente pensaram nisso..." (Docente 01).*

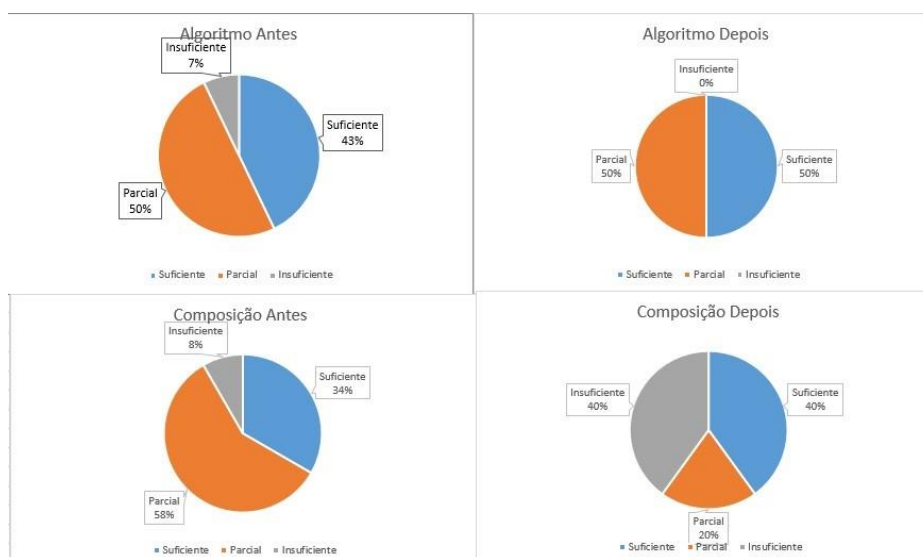
*"Nos ajuda a pensar em estratégias mesmo! Por exemplo, a questão do perceber os padrões, como isso é super pratico quando a gente vai aplicar, tipo a gente entende quando a gente faz um quiz, a gente faz um exercício e a gente sabe que existe um padrão, ainda que algumas coisas mudem, coisas se repetem e aquilo que aparece na teoria, na prática fica muito evidente". (Igor).*

### 4.5.2 Intervenção 02

As habilidades de Algoritmo e Composição e Decomposição foram desenvolvidas na Intervenção 02. Na aptidão de Algoritmo houve um crescimento no índice de satisfação em queda ao de insuficiente. A prática de decomposição e composição houve um retrocesso nos valores iniciais da classe parcial e uma crescente na classe insuficiente conforme Figura 11. Já a segunda intervenção apresentou resultados positivos na compreensão dos algoritmos e comunicação entre docente e discente.

*"Você apresentou conceitos, que ao decorrer do curso, por exemplo; você explicou o que era um algoritmo, explicando mais aprofundado". (Docente 01).*

**Figura 11 – Comparação após a intervenção 02**



Fonte: Próprio autor.

### 4.6 Resultados Individuais

Ao final da pesquisa pode-se desenhar um caminho de cada participante com base nos dados coletados. Seguem registros cronológico dos indivíduos, os quais foram gerados com base nos dados coletados e a avaliação final deles. Lembrando que no momento desta pesquisa, alguns alunos estavam pendentes da entrega de seus trabalhos finais, o que não possibilitou a conclusão cronológica deles.



Bernardo: Iniciou com os pilares abstração e reconhecimento de padrões classificados por insuficiência, algoritmo encontrava-se parcial e sua decomposição como suficiente. Durante as intervenções ele continuou apresentando abstração insuficiente e conseqüentemente as habilidades que dependência de usar uma maior abstração ele apresentava dificuldades. Apresentou uma melhora no pilar de reconhecimento de padrões e manteve os outros demais. Ao final do curso não apresentou melhora significativas nos pilares, mantendo todos na mesma categoria.

Gabriel: Inicialmente não conseguiu apresentar abstração durante a atividade o que a categorizou como insuficiente, seu reconhecimento de padrões, algoritmos e decomposição foram parciais. Ao decorrer do curso conseguiu elevar todas as suas habilidades para suficiente. Mostrou-se interessado, sempre tirando dúvidas, além de ser bastante participativo em sala de aula.

Igor: Começou com todas as habilidades categorizadas como suficiente. Manteve-se assim durante todas as intervenções. Programa de acordo com o que foi abordado em sala de aula.

Aparecida: Chegou até o curso com as habilidades de abstração e reconhecimento de padrões classificadas como insuficiente. Suas habilidades de decomposição e algoritmo foram classificadas como parcial. Durante as intervenções ela mostrou evolução nos pilares de Abstração e Reconhecimento de padrões como parciais. Ao final do curso apresentou um domínio sobre os conteúdos que foram apresentados.

Esther: Possuía habilidade de abstração foi categorizada como suficientes, porém suas habilidades de algoritmos foram parciais e reconhecimento de padrões e decomposições foram classificadas como insuficientes. Melhorou os pilares de Reconhecimento de Padrões e Algoritmo. Participativa, dominou os conteúdos apresentados sem muitas dificuldades.

Aurora: Apresentou habilidades iniciais insuficientes para abstração e reconhecimento de padrões. Algoritmo e Decomposição foram categorizado como parciais. No decorrer do curso todas as quatro habilidades evoluíram. Ao final apresentou um bom entendimento dos pilares.

Amanda: Referente a avaliação das habilidades iniciais dela no que tange o raciocínio computacional foi observado que o reconhecimento de padrões se encontra em nível parcial, quanto a algoritmos e abstrações estão em níveis suficientes, porém a

decomposição foi classificada como insuficiente. Sua interpretação decomposição melhorou durante o curso.

## 5. Conclusões e Trabalhos futuros

O desenvolvimento do presente estudo foi de grande importância para ampliar o conhecimento do autor sobre o tema, possibilitou ao mesmo uma análise de como o raciocínio computacional pode ajudar a melhorar a aprendizagem de iniciantes em programação. Além disso, também permitiu conceber uma proposta para um melhor desempenho destes iniciantes.

Tedesco *et al.* (2016) fala que as dificuldades encontradas pelos discentes quando iniciantes em programação, fruto do processo de aprender a programar pode afetar negativamente como eles se sentem capazes de aprender a programar e como consequência antecipar o insucesso. Esta negativa pode gerar uma evasão da área, uma vez que o principiante se acha incapaz de programar. Assim, reforça-se a importância do tema apresentado nesta pesquisa não só para os novatos em programação, mas também aos que já possui alguma experiência com a programação.

Com este trabalho foi constatado que até mesmo intervenções breves, como as utilizadas nesta pesquisa, quando elaboradas com uma metodologia adequada, podem produzir efeitos relevantes para seus participantes. O fato de duas intervenções de uma hora de duração cada, promoveram mudanças na capacidade de aprendizado, possíveis de serem mensuradas pelos docentes e alunos e serem fortemente relatados em seus depoimentos, confirma a importância de se inserir o Raciocínio Computacional na introdução a programação.

Outro dado importante foi a redução nos valores dos classificados como insuficientes nas habilidades de Abstração, Algoritmo e reconhecimento de Padrões quando comparado ao início do curso. Cabe destacar, que as habilidades do raciocínio computacional podem e são utilizadas em diversas áreas e não apenas em computação, como foi relatado por alunos que utilizaram das mesmas habilidades para atividades em seus trabalhos/cursos.

## 5.1 Trabalhos Futuros

Para concluir, a partir dos conteúdos desenvolvidos por este estudo, é possível notar como as habilidades do Raciocínio Computacional alinhado aos conceitos de programação contribuem na aprendizagem dos estudantes iniciantes em programação. Estes conteúdos constroem novas linhas de pesquisa sobre o tema. Levando em consideração essas linhas de pesquisa, pode ser aplicado os conceitos discutidos nesta pesquisa com os ingressantes nos Cursos de Computação das Universidades, e confrontar os resultados obtidos neste estudo, com grupos de iniciantes que não tiveram contato com o raciocínio computacional e com grupos que obtiveram um tempo maior de contato com as habilidades de raciocínio computacional.

## 6. Referências

- BARDIN, L. **Análise de conteúdo (edição revista e atualizada)**. Lisboa: Edições, v. 70, 1995.
- BEAUBOUF, T.; MASON, J. Why the high attrition rate for computer science students: some thoughts and observations. **ACM SIGCSE Bulletin**, ACM, v. 37, n. 2, p. 103–106, 2005.
- BIGGS, J.; TANG, C. **Teaching for quality learning at university mcgraw-hill international**, 2011.
- CASPERSEN, M. E.; KOLLING, M. Stream: A first programming process. **ACM Transactions on Computing Education (TOCE)**, ACM, v. 9, n. 1, p. 4, 2009.
- CRESWELL, J. **Qualitative inquiry and research design. choosing among five approaches** (3 ed.). London: Sage. Approches inductives, 2013.
- GIL, A. C. **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 2002, 192 p.
- GOMES, A. J.; SANTOS, A. N.; MENDES, A. J. A study on students' behaviours and attitudes towards learning to program. In: **Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education**. [S.l.], 2012. p. 132–137.
- KITCHENHAM, B. Procedures for performing systematic reviews. **Technical Report TR/SE-0401** Keele, UK, Keele University, v. 33, n. 2004, p. 1–26, 2004.
- KURLAND, D. M.; PEA, R. D.; CLEMENT, C.; MAWBY, R. A study of the development of programming ability and thinking skills in high school students. **Journal of Educational Computing Research**, SAGE Publications Sage CA: Los Angeles, CA, v. 2, n. 4, p. 429–458, 1986.
- LAHTINEN, E.; ALA-MUTKA, K.; JÄRVINEN, H.-M. A study of the difficulties of novice programmers. **ACM SIGCSE Bulletin**, v. 37, n. 3, p. 14–18, 2005.
- LEARNING, B. B. **What is computational thinking?** 2018. Disponível em: <<https://www.bbc.com/education/guides/zp92mp3/revision/1>>. Acesso em: 20 jul. 2018.
- LIUKAS, L. **Hello Ruby: adventures in coding**. [S.l.]: Macmillan, 2015. v. 1.
- PERKINS, D.; MARTIN, F. Fragile knowledge and neglected strategies in novice programmers. In: **First workshop on empirical studies of programmers on Empirical studies of programmers**. [S.l.: s.n.], 1986. p. 213–229.
- PROULX, V. K. Programming patterns and design patterns in the introductory computer science course. **ACM. ACM SIGCSE Bulletin**. [S.l.], 2000. v. 32, n. 1, p. 80–84.
- RIBEIRO, L.; FOSS, L.; CAVALHEIRO, S. A. C. **Entendendo o pensamento computacional**. ArXiv Preprint ArXiv:1707.00338, 2017.

TEDESCO, P. C. A. R.; ANDRADE, W. L.; MATOS, E. S. **Um Estudo sobre o Desenvolvimento de Interesse pela Aprendizagem de Programação**. SCAICO, P. D. (Orientador), 2016.

SHEARD, J.; SIMON, S.; HAMILTON, M.; LÖNNBERG, J. Analysis of research into the teaching and learning of programming. In: **Proceedings of the fifth international workshop on Computing education research workshop**. [S.l.], 2009. p. 93–104.

SPOHRER, J. C.; SOLOWAY, E. Simulating student programmers. **Ann Arbor**, vol. 1001, p. 1-48109, 1989.

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

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

## APÊNDICES

### APÊNDICE A – Termo de Consentimento Livre e Esclarecido

Universidade Federal da Bahia

Instituto de Matemática e Estatística

Departamento de Ciência da Computação

Grupo de Pesquisa e Extensão em Informática, Educação e Sociedade - Onda Digital

#### TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

O/A senhor(a) está sendo convidada(o) a participar voluntariamente de uma pesquisa sobre “Como o raciocínio computacional pode ajudar a melhorar a aprendizagem de iniciantes em programação?”. Essa pesquisa tem o objetivo de identificar possíveis relações entre as habilidades do raciocínio computacional desenvolvidas pelos iniciantes em programação e sua melhora na aprendizagem. Nesse estudo de caso, os participantes terão que responder um questionário e resolver algumas atividades. O questionário contém perguntas relativas ao seu interesse pelo curso de programação, bem como sua motivação.

Esses procedimentos não oferecem risco algum à integridade física e moral dos seus participantes, bem como despesas, prejuízos ou benefícios diretos. Para participar deste estudo o/a senhor(a) não terá nenhuma despesa, nem receberá qualquer vantagem financeira. Caso o/a senhor(a) tenha alguma dúvida ou necessite de qualquer esclarecimento ou, ainda, deseje retirar o consentimento de participação da pesquisa, por favor, entre em contato com os pesquisadores abaixo a qualquer tempo.

Pesquisador responsável: Danilo Santos dos Santos,

danilosantos@dcc.ufba.br,

(71) 9XXXX-XXXX

**APÊNDICE B – Questionário de Perfil**

## 1. Alguns dados sobre você

1.01 - Qual é o seu nome completo?

1.02 - Qual é a sua idade?

1.03 - Qual é o seu e-mail?

1.04 - Qual é a sua turma?

## 2. Dados sobre interesses e escolhas pessoais

2.01 - O que lhe motivou a escolher esse curso de Programação?

 Recomendação familiar. Tenho parentes que trabalham nessa área. Pelo incentivo ou exemplo de um(a) amigo(a). Pelo incentivo ou exemplo de um(a) professor(a). Já possuía interesse na área. Possível retorno financeiro. Outros.

2.03 - Você já teve algum contato com programação antes desse curso?

 Sim, já fiz outra faculdade/curso onde estudei programação. Sim, através de um(a) amigo(a). Sim, por algum conteúdo na internet relacionado. Sim, em outro contexto/lugar. Sim, foi necessário para meu trabalho atual/anterior. Não.

2.04 - Por favor, caso tenha informado "Sim, em outro contexto/lugar na questão anterior, informe esse contexto.

## APÊNDICE C – Características dos Participantes

### Quadro 3 – Características dos participantes do estudo

#### Legenda CP: Contato com Programação

Nome	Idade	Turma	CP	Descrição
Aurora	30	A	Não	Aurora não possui contato prévio com programação, mas buscou o curso por interesse acadêmica. A mesma apresentou habilidades iniciais insuficientes para abstração e reconhecimento de padrões. Sua habilidade de algoritmo e sua decomposição foram categorizado como parciais.
Aparecida	19	A	Não	Chegou até o curso por interesse na área e incentivo de amigo, a mesma não tinha contato com programação. Suas habilidades de abstração e reconhecimento de padrões foram classificados como insuficiente. Suas habilidades de decomposição e algoritmo foram classificadas como parcial.
Agatha	27	A	Não	Possui interesse na área e não teve nenhum contato com programação. Sua abstração e reconhecimento de padrões são parciais. Suas habilidades de algoritmos e decomposições foram consideradas suficientes.
Esther	20	A	Não	Foi motivada por um exemplo ou incentivo de um amigo, mas não tinha contato com programação antes. Sua habilidade de abstração foi categorizada como suficientes, porém suas habilidades de algoritmos foi parcial e reconhecimento de padrões e decomposições foram classificadas como insuficientes.
Amanda	30	A	Não	Possui interesse na área e nenhum contato com programação. Referente a avaliação das habilidades iniciais dela no que tange o raciocínio computacional foi observado que o reconhecimento de padrões encontra-se em nível parcial, quanto a algoritmos e abstrações estão em níveis suficientes, porém a decomposição foi classificado como insuficiente.
Cláudia	18	A	Não	Possui interesse na área e contou com incentivo ou exemplo de um amigo para começar o curso, seu contato com programação segundo a mesma é nenhuma. Ela apresentou abstração e reconhecimento de padrões parciais, suas habilidades de decomposição e algoritmos também.
Vanessa	18	A	Não	Buscava aprender mais sobre a área, pois possuía interesse na área e foi incentivada por amigo, mas não teve contato inicial com programação antes do curso. Sua abstração foi suficiente para responder o questionário, porém seu reconhecimento de padrões foram parcial, assim como algoritmo e decomposição.



Gabriela	18	A	Não	Possuía interesse na área e foi incentivada por um(a) amigo(a) a fazer o curso. Não tinha contato inicial com programação. Sua abstração e reconhecimento de padrões foram parciais. Enquanto que decomposição e algoritmos foram classificados como insuficientes.
----------	----	---	-----	---

Fonte: Próprio autor.

**APÊNDICE D – Características dos Participantes Turma B****Quadro 4 – Características dos participantes do estudo****Legenda CP:** Contato com Programação

Nome	Idade	Turma	CP	Descrição
Bernardo	23	B	Sim	Possui parentes que trabalham na área e por recomendação da família escolheu o curso, o mesmo havia tido contato antes com programação por meio de conteúdo na internet. Nas habilidades de abstração e reconhecimento de padrões ele foi categorizado como Insuficiente nas atividades propostas. Sua habilidade de algoritmo foi desenvolvida parcialmente e sua decomposição foi suficiente
Gabriel	26	B	Sim	Buscou o curso por interesse na área e possível retorno financeiro e já possui contato com programação através de um curso que iniciou mas não terminou. Gabriel não conseguiu apresentar abstração durante a atividade o que a categorizou como insuficiente, seu reconhecimento de padrões, algoritmos e decomposição foram parciais.
Vicente	53	B	Sim	Chegou até o curso por interesse na área, o mesmo teve contato com programação por necessidade do seu trabalho. Nas habilidades de abstração, reconhecimento de padrões e decomposição ele conseguiu interpretar parcialmente a demanda. Sua habilidade de algoritmo foi suficiente.
Augusto	23	B	Sim	Chegou ao curso por possui interesse na área, o mesmo informou que já tinha feito curso de graduação em redes de computadores e pegou uma matéria de programação, o mesmo trancou o curso. Das habilidades apresentada apenas a de algoritmos apresentou categorização como suficiente. Decomposição e abstração ele interpretou parcialmente e reconhecimento de padrões ficou insuficiente.
Igor	31	B	Sim	Entrou no curso por uma necessidade acadêmica (grupo de pesquisa) e já possuía contato com programação através de conteúdos na internet. O mesmo apresentou em todas a habilidades uma categorização suficiente.
Catarina	25	B	Não	Não possuía nenhum contato antes com programação e o que a levou ao curso foi que a mesma estagiava em um laboratório que trabalha com programação. A mesma não conseguiu interpretar as habilidades propostas o que categorizou suas habilidades como insuficientes.
Natália	38	B	Não	Chegou ao curso por interesse na área, porém não possui nenhum contato com programação. Ela apresentou uma categorização como suficiente para todas as habilidades propostas.

Fonte: Próprio autor.

## **APÊNDICE E – Entrevista Docente 01**

### **ENTREVISTA INSTRUTORA 01**

Pesquisador: Inicialmente queria que você me contasse um pouco sobre o cargo de professora, qual o curso faz se já havia ensinado outras turmas ou não, é a primeira turma? Se tem experiência em dar essas aulas de Ciência da Programação?

Docente 01: Certo eu sou do Bacharelado da disciplina Ciência da Tecnologia, como estrutura durante o semestre inteiro é a primeira vez que fico com uma turma. No semestre passado eu fiquei compartilhando aulas com outra pessoa então uma complementava a outra e eu estava mais durante as aulas de exercício. É.. Desde 2016 eu já estava ajudando com monitoria e a experiência de ir pra sala de aula pra tá com os alunos na frente de uma turma foi algo diferente, porque tem relação com o curso futuro que eu pretendo que é Licenciatura em Computação.

Pesquisador: Qual expectativa você tinha quando iniciou com a turma? De acordo com o perfil que vocês fizeram a seleção? Você fez a seleção você criou um perfil de alunos que tinha uma expectativa, no caso com as meninas assim, qual sua expectativa quando iniciou?

Docente 01: Olha, durante o processo de seleção as meninas eu não estava! Então, esse perfil foi composto por outras pessoas e alguns outros membros, mas assim da primeira aula que eu tive contato com elas eu achei que a turma conseguiria avançar um pouco mais do que elas me apresentaram, agora no final. Por exemplo: elas tiveram muitas dificuldades no começo do curso que eu achei que não seria fatores que irão é.... intervir no processo de aprendizagem delas e que em certos momentos mostrou muito lento por conta até, acho dos espaços iniciais com relação a lógica.

Pesquisador: E quanto as intervenções o que você achou das aulas em si? Foi bom, foi ruim? O que você tem de sugestões ou melhoria a ser fazer nesse sentido das intervenções que eu fiz?

Docente 01: Certo! Olha, eu achei que foi algo que realmente agregou porque durante muitas aulas após ter feito intervenções elas usaram exemplos das coisas que tinha acabado de falar durante a aula. Por exemplo: Fica o entendimento de algum conteúdo. Agora eu acho que mesmo se tratando de TCC ou uma Pesquisa poderia ter um elemento de interação maior com elas. Algo que estimulasse a não somente responder as questões.

Uma forma não sei, se seria lúdica, mas algo que atraísse mais fácil a atenção delas. Porque pelo menos o que eu conseguir perceber é que elas se sentem um pouco desafiada a fazer as coisas, então é um estímulo quando você leva isso pra uma turma. Pesquisador: E você tem em si o raciocínio computacional, o que você entende de raciocínio sobre o raciocínio computacional:

Docente 01: No fundo eu nunca tive um real entendimento. O que eu conheço é o que eu presenciei em algumas palestras e de conversa com outros grupos que trabalham com a temática. Eu sei que envolve muito mais do que você pensar somente com o computador envolve você ter toda uma estratégia lógica por traz e isso é que compõe o que a gente conhece como computador.

Pesquisador: Entendi! E você acha isso bom, ruim? Sua opinião é algo útil, não é? Pra uma turma de iniciação de programação.

Docente 01: É essencial como falei antes elas mostram dificuldades com coisas básicas de lógica, então, quando você consegue trabalhar com o aluno questão de lógica eles não levam somente pra programação eles conseguem aplicar em outros setores, e eles pensam nessa estrutura até mesmo quando eles partem pra outras matérias. Ex: Tinha duas alunas do ensino médio se eu não me engano e isso pode fazer um grande diferencial pra elas.

Pesquisador: Você aplica ou já aplico o raciocínio computacional em alguma turma que você já ensinou ou algo do tipo que você acha que é raciocínio computacional?

Docente 01: É isso, eu não saberia classificar se realmente eu fiz isso, se dinâmicas estiverem relacionadas, dinâmicas e uso de computador estiverem relacionadas eu acho que sim!

Pesquisador: E dentro do que você acompanha deles o que você acha? Qual a visão de mundo sobre o raciocínio computacional depois das intervenções?

Docente 01: Eu acho que quando você conseguiu suas intervenções com ela sem que os conceitos fossem apresentados de um modo tão formal. Isso conseguiu fazer o diferencial para o entendimento delas, tanto é, que elas conseguiram levar pra aulas, e eu creio que realmente ajudou e mudou a visão que elas tinham do curso que só entende o que é fazer um programa de só entender o que é sentar e passar algo pra C.

Pesquisador: Agora, um comparativo da turma anterior que você falou que tinha lecionado que não teve contato com essa noção de raciocínio computacional e essa turma você classificar como essa comparação se foi bom ou ruim?

Docente 01: Eu acho que a maior diferença é na consistência que elas sintetizaram apesar de eu considerar que a gente fez um bom trabalho nas outras turmas nessa a gente pode considerar que o que elas aprenderam não ficou só no papel eu vejo que se o aluno consegue pegar aquilo que você passou do conceito e ver a aplicação em outros locais em outras formas já fez sentido.

Pesquisador: Agora, tipo um exemplo: teve algum exemplo ou situação que na turma passada você teve alguma dificuldade pra passar o assunto e nessa foi mais fácil por causa dessa intervenção que você recebeu? Algum exemplo tipo, a eu ensine loop A turma passada teve dificuldade, nessa não.

Docente 01: Logo nós assunto iniciais de if. Onde você conseguiu fazer a intervenção sobre abstração e elas realmente pensaram nisso no momento de olhar pra o problema e tentar solucionar foi uma coisa que marcou por que elas realmente usaram o termo abstrair pra solucionar o problema.

Pesquisador: Uma escala de 0 a 10 qual nota você dar no desenvolvimento da sua turma?

Docente 01: Eu acho que seria 8, por que eu confesso que também houve a parte que gente ficou por conta das pausas que a gente teve a gente só tem aula uma vez por semana então a gente não conseguia ter o ritmo que elas fizessem os exercícios e levassem dúvidas e conseguissem elaborar. Muitas tinham aula durante a semana atrapalhava e então já chegavam no sábado sobrecarregadas do que realmente estavam com dúvidas do assunto.

Pesquisador: E você acha que por exemplo sua turma fosse dividida em dois dias na semana agregaria mais valor?

Docente 01: Sim, com certeza! Porque teria esse momento pra eles, realmente de pensarem mais no assunto e aí você com esse intervalo de pausa pra refletir e com 3 horas seguidas de aula a gente leva o conteúdo todo. Então você vai ter conteúdo, vai ter exercício e naquele momento vai ficar com dúvidas se realmente entendi ou não!?

Pesquisador: Uma dúvida final, você acha que antes das aulas de programação, assim, exemplo: teria que ter duas, três, quatro aulas só de raciocínio computacional pra preparar essa turma pra programação ou você acha que não?

Docente 01: Sim, porque com o ensino computacional elas conseguiriam desenvolver melhor as concepções delas lógicas e é uma coisa que como elas não conseguiram progredir bem no início do curso até porque tivemos essas pausas, agora elas apresentaram um pouco de dificuldade com a sequência pra resolver um problema, entender a parte do

problema, elas sabem até fazer mas não sabem como estruturar aquilo logicamente. Pesquisador: Tá! E pra finalizar dessa 4 habilidades de decomposição /composição, abstração, reconhecimento de padrões e algoritmo. Qual você achou de maior importância no desenvolvimento dela e qual você achou de menos importância pra equilibrar pra que a gente pudesse balancear no futuro.

Docente 01: Eu não diria de menor relevância por que eu não consigo ver nenhuma que não tenha nenhuma relevância elas não podem ter dito diretamente sobre o conteúdo, mas terem percebido o reconhecimento do conteúdo que você passou eu acho que as que mais foram notadas, foi o algorítmico e abstração foi realmente o que elas citaram então não teria algo que foi menos importante acho que em si os quatro conseguem formar uma concepção total.

**APÊNDICE F – Entrevista Docente 02****ENTREVISTA INSTRUTORA 02**

Pesquisador: Curso?

Docente 02: Ciência da Computação

Pesquisador: Idade?

Docente 02: 25 anos.

Pesquisador: Já lecionou em outras turmas?

Docente 02: Já. Não de programação, mas eu já fui monitora de outras turmas de iniciação à informática e já dei aula que não era bem uma turma, porque era um público rotativo no Grupo de Apoio à Crianças com Câncer.

Pesquisador: E você lecionou na turma sozinha?

Docente 02: Sendo monitora?

Pesquisador: Sim, mas responsável, sozinha.

Docente 02: Sim, no semestre passado que eu fiquei com a turma de iniciação à informática e nesse semestre que eu fui responsável pela turma, mas em outros cursos eu fiquei de monitora (auxiliar), só.

Pesquisador: Sendo assim, tem quanto tempo você lecionando, contato tudo? Docente

02: Aqui no Onda? Três. Que é só no Onda mesmo, que eu sou aula.

Pesquisador: Já tinha dado aula de iniciação à programação antes?

Docente 02: Como monitora, não como responsável da turma pra ficar à frente da turma, é a primeira vez.

Pesquisador: [Pergunta inaudível]

Docente 02: Foi 2017.1, foram três meses o curso.

Pesquisador: Quando é feito o processo de seleção, vocês criam um perfil do aluno, não sei se você participou. Qual foi sua expectativa no início, quando você viu o perfil de cada aluno? Ou você não teve acesso a esses dados?

Docente 02: Acesso aos dados em si, no apanhado geral não, mas eu pude acompanhar por que, durante o processo de seleção, eu participei da entrega de documentos, aí é onde

há a aplicação ver qual o nível. Eu não sei meus alunos, por que eu não tive acesso ao deles.

Pesquisador: Então você acabou não criando expectativa deles?

Docente 02: Não e os que eu pude ver, eles não tinham noção do que era programação.

Pesquisador: Você conhecia a raciocínio computacional antes, não?

Docente 02: Sim

Pesquisador: Já aplicou, ou usou? O que você conheceu da raciocínio computacional?

Docente 02: Raciocínio computacional, que você fala, é ligado à lógica...Então, só o que eu aprendi na sala de aula mesmo, o que aprendi durante o curso.

Pesquisador: Então esse conceito você aprendeu durante o curso mesmo?

Docente 02: Não, foi durante o meu curso de ciência da computação.

Pesquisador: É isso, mas o raciocínio computacional no sentido de que [parte inaudível]?

Docente 02: Não, ninguém nunca me apresentou dessa forma.

Pesquisador: Você já aplicou essas intervenções antes?

Docente 02: Não, como essa foi a primeira turma a intervenção foi só sua, não tinha visto.

Pesquisador: Sua turma desse semestre, como eles começaram em questão de conhecimento de programação?

Docente 02: A turma no caso que começou com oito pessoas e agora ao final chegaram três. No começo eles não tinham muita noção, de...Na verdade, eles não tinham noção de algoritmo, da linguagem C, de lógica e com o decorrer do curso e tal, eles apresentaram um avanço, pude perceber um avanço, e os que restaram né? Os que no caso vão se certificar no curso, eu gostei do resultado deles, porque desde o começo, foram os que se destacaram mesmo, foram os que faziam as atividades, os que tiravam dúvida, os que chegavam com questionamento, que paravam para me perguntar no WhatsApp. Então, eu acho que o nível da turma, no caso deles, foi por conta do interesse deles. Então eu acho que eles tiveram um desempenho melhor, por que eles se dedicaram mais, pelo menos, em relação a mim, foi o que eu pude perceber. Foram os que se dirigiram a mim, tiravam dúvidas, então pude perceber que eles...

Pesquisador: Eles não tinham conhecimento de programação...?

Docente 02: Não, nunca nem tinham visto.



Pesquisador: Você percebeu alguma dificuldade deles no curso, qual assim você pode destacar?

Docente 02: A dificuldade foi por que eles não tinham mesmo noção da lógica, por que a linguagem a gente ensina, mas acho que a linguagem ela só vai diferenciar mesmo o escopo da linguagem, mas a lógica mesmo por trás, eles não sabiam também, não tinham noção da lógica

Pesquisador: E nesse processo, qual foi o desempenho deles na iniciação?

Docente 02: Assim, pegando de onde eles começaram até agora, acho que o desempenho deles foi relativamente bom pra um curso de programação, para que não havia visto nada, Porque eu mesma, quando comecei eu acho que demorei mais para entender do que eles. Eu não sei se talvez pela forma de explicar que eu não acompanhei, assim da forma que passaram pra mim foi diferente aí eu aprendi para não passar pra outras pessoas daquela forma, então eu não sei se tem a ver com isso, mas em relação de como começaram ao que estão agora, eu acho que eles evoluíram muito.

Pesquisador: Então, eu apliquei duas intervenções nas turmas e foram duas intervenções e logo depois você continua a aula, você acha que essas intervenções ajudaram ou atrapalharam?

Docente 02: Eu acho que ajudaram sim, por que você apresentou conceitos que, por exemplo, você chegou explicando o que era um algoritmo, explicando mais aprofundado coisa que a gente não tinha tempo hábil pra aprofundar, a gente mostrava o que era e o que fazia, mas você chegou e aplicou mais detalhadamente, ensinou a eles mais detalhadamente e é como eles mesmos falaram né? Eles falaram que as intervenções tiveram relevância neles, inclusive teve um que disse que são conceitos que ele vai levar pra vida, então é só em sala de aula que as intervenções fizeram diferença pra eles, mas no pessoal deles eu acho que fizeram

Pesquisador: Você consegue destacar assim, um momento dessas intervenções que você notou um conceito ter ficado mais fácil de ser replicado, ou seja, houve um antes e depois na forma de passar por ter facilitado.

Docente 02: O conceito de algoritmo eu acho que não expliquei como deveria, você explicou mais aprofundado e com isso eu acho que eles desenvolveram mais, assim quando eles tinham uma dúvida eles sabiam explicar o que era aquela dúvida, eles já sabiam

identificar e dizer o que era o problema de uma forma que eu entendesse, então eu acho que foi bastante relevante nesse sentido.

Pesquisador: Você já teve turma que foi monitora, houve uma dificuldade que teve nessas turmas passadas de programação como monitora e agora como responsável não teve mais, por exemplo, você ensinando FOR, na outra turma teve uma dificuldade que nessa já não teve após as intervenções?

Docente 02: Entendi, mas eu não consegui identificar por que na verdade quem dava as aulas mesmo não era eu, eu só ficava, corrigindo atividades, tirava dúvidas, então eu não sei se teve alguma diferença.

Pesquisador: Numa escala de zero à dez como você avalia o desempenho da turma, para o que você esperava dela?

Docente 02: Aos que restaram, eu daria oito, assim eu não vou dá dez, porque tiveram coisas que não que não supriram as expectativas, pois achei que foi bastante satisfatórios, mas acho que pro dez ainda falta alguma coisa, mas em relação ao que eu esperava por ser a minha primeira turma, por eu não ter tanta experiência assim em ensinar programação eu acho que foi bem satisfatório.

Pesquisador: A última pergunta é referente às intervenções assim, você acha que elas foram úteis assim, nas próximas turmas deveriam trabalhar elas, nas próximas atividades elas deveriam fazer parte do programa do curso? [Não foi possível ouvir toda pergunta]

Docente 02: Eu vou falar assim, mas nem é uma opinião minha, mas o que os alunos disseram, e se tipo, eles que tiveram a intervenção e eles disseram que foi relevante pra eles e que nas próximas turmas seria bem legal se tivesse na ementa do curso, eu acho que deveria sim, abordar mais profundamente na ementa do curso. Pesquisador: Você bolsista do Onda Digital?

Docente 02: Sim

Pesquisador: Obrigado!

## APÊNDICE G – Atividade Habilidades Abstração e Reconhecimento de Padrões

### Quiz Abstração e reconhecimento de padrões

\*Obrigatório

1. Nome \*

---

2. Turma \*

---

#### Perguntas do quiz

3. 1. Das situações abaixo, qual apresenta uma demonstração do reconhecimento de padrões? \*

*Marcar apenas uma oval.*

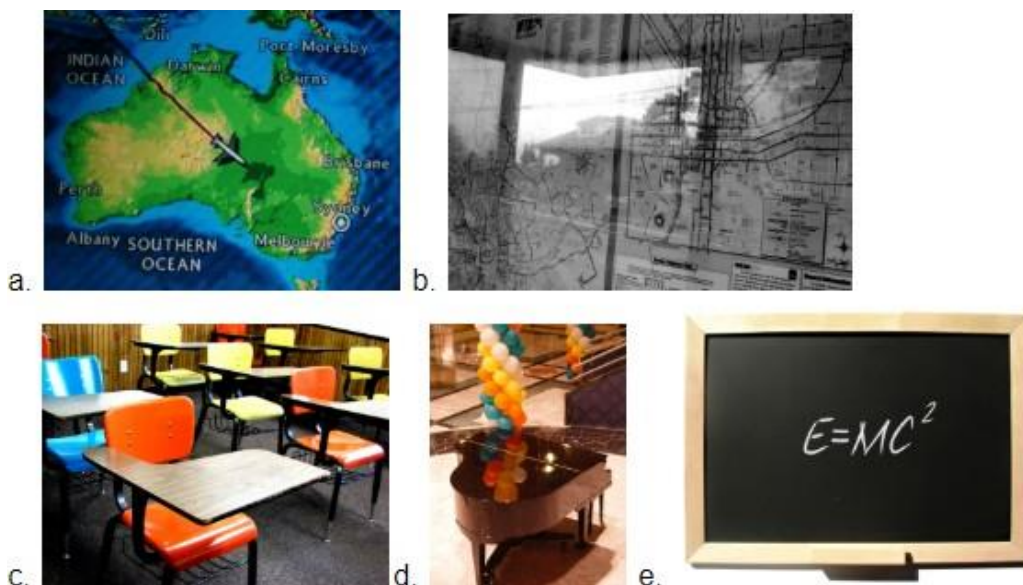
- Um biólogo classifica espécies de aranhas, baseado em suas cores;
- Ao tentar anexar um arquivo a um e-mail, o aluno percebe que tal ação só será possível se o arquivo passar por um programa de compressão.
- Um programador desenvolve um algoritmo que reconhece rostos em uma foto
- Um músico compoendo parte de uma melodia.

4. 2. Sobre reconhecimento de padrões, marque V (Verdadeiro) ou F (Falso): \*

*Marcar apenas uma oval por linha.*

	Verdadeiro	Falso
Sendo uma habilidade da Ciência da Computação, o reconhecimento de padrão não pode ser utilizado fora de sua área de origem;	<input type="radio"/>	<input type="radio"/>
Desbloquear um computador por comando de voz, envolve reconhecimento de padrões	<input type="radio"/>	<input type="radio"/>
O reconhecimento de padrões contribui na economia de tempo, pois permite a repetição de operações.	<input type="radio"/>	<input type="radio"/>
É possível perceber o reconhecimento de padrões na escala de notas musicais	<input type="radio"/>	<input type="radio"/>

5. 3. Das imagens abaixo, quais você consideraria como um exemplo de abstração? \*  
imagens retiradas do site <http://www.freeimages.com/> \*



Marcar apenas uma oval.

- Apenas a, b, e
- Apenas a, c, e
- Apenas d, c, e
- Apenas c, b, d
- Apenas a,b,d

6. 4. Das situações descritas abaixo, qual não se adequa ao conceito de padrão? \*

Marcar apenas uma oval.

- O pôr do sol e o nascer do sol
- As estações do ano
- O filtro antispam de um seu e-mail
- Um sorteio da loteria

7. 5. Abstração e reconhecimento de padrões são habilidades importantes na resolução de problemas. Você consegue relacionar o uso de tais habilidades com sua área de atuação? Em quais momentos? \*

---



---



---



---



## APÊNDICE H – Atividade Habilidades Decomposição e Algoritmos

### Quiz Abstração e reconhecimento de padrões

\*Obrigatório

1. Nome \*

2. Turma \*

### Perguntas do quiz

3. 1. Das situações abaixo, qual apresenta uma demonstração do reconhecimento de padrões? \*

Marcar apenas uma oval.

- Um biólogo classifica espécies de aranhas, baseado em suas cores;
- Ao tentar anexar um arquivo a um e-mail, o aluno percebe que tal ação só será possível se o arquivo passar por um programa de compressão.
- Um programador desenvolve um algoritmo que reconhece rostos em uma foto
- Um músico compondo parte de uma melodia.

4. 2. Sobre reconhecimento de padrões, marque V (Verdadeiro) ou F (Falso): \*

Marcar apenas uma oval por linha.

	Verdadeiro	Falso
Sendo uma habilidade da Ciência da Computação, o reconhecimento de padrão não pode ser utilizado fora de sua área de origem;	<input type="radio"/>	<input type="radio"/>
Desbloquear um computador por comando de voz, envolve reconhecimento de padrões	<input type="radio"/>	<input type="radio"/>
O reconhecimento de padrões contribui na economia de tempo, pois permite a repetição de operações.	<input type="radio"/>	<input type="radio"/>

É possível perceber o  
reconhecimento de padrões na  
escala de notas musicais

---



5. 3. Análise a imagem abaixo e selecione as frases abaixo que são verdadeiras: \*



*Marque todas que se aplicam.*

- a. Um bibliotecário utiliza habilidades computacionais para administrar a biblioteca.
- b. Para melhor encontrar um livro dentro da biblioteca dividimos ela em sessões.
- c. Essa biblioteca é uma composição de material escrito e estantes.
- d. A melhor forma de organizar os livros na estante é de forma aleatória.

6. 4. Quanto à questão anterior (3), diga qual habilidade é utilizada em alternativa e rescreva as alternativas falsas de forma a tornarem-se verdadeiras e dizer quais as habilidades: \*

---

---

---

---

---

Powered by

## APÊNDICE I – Grupo Focal

### GRUPO FOCAL

Pesquisador: Pra primeira pergunta que tenho pra vocês é: O que é que vocês acharam das intervenções? Aquelas duas intervenções, que é que vocês acharam? [Bernardo] Eu achei algo interessante, algo produtivo, que a gente pode ter o conhecimento a mais e também pra assim, como solucionar umas coisas “praticante” já tem o nome né? É só...A gente não tem nome definido, né? Mas tem o

Pesquisador: Vocês também pode ficar a vontade de falar, pode completar a fala. [Gabriel] Eu achei assim pra nós, assim pra nós não, pra mim vou falar por mim, pra nós que não viemos desse lugar, da ciência da computação, da programação, a nos ajuda a se familiarizar, com alguns conceitos, alguns termos usados e que dizem respeito a prática em si, lá do... Muitas dessas intervenções terem assim...serem teóricas né? Abstração da prática ela acaba nos ajudando num pouco do que ela falou assim, no sentido de nomear coisas que a gente faz, que a gente as vezes não entende bem né?

[Igor] Eu concordo, eu achei interessante também, justamente por ter conseguido trazer alguns conceitos, que são subjacentes a questão da programação e ao mesmo tempo consegui demonstrar pra gente que são coisas que a gente meio que aplica no dia-a-dia em outras tarefas, outras atividades, mas que fazem parte da atividade de programar. Pesquisador: Vocês acharam que essas intervenções, elas foram úteis pra vocês durante o curso? Não precisa ser todas as 4. E em que sentido vocês lembram e dizem “isso foi muito útil nesta parte aqui”?

[Gabriel] Programação é isso, é você ter uma parte, né? O programa quando

[Inaudível2:35-2:40] e aí? Começar do começo e ai desvendar [inaudível do 2:43 2:50] [Igor] Pra mim foi bem assim, nesse sentido assim, de nos ajudar a nomear coisas da prática não só nomear, mas também a entender processos da prática de programação, o próprio conceito de algoritmo é...quando a gente tá realmente efetivamente programando é interessante pensar assim que, para compor um algoritmo a gente tem que pensar em etapas e que começa num lugar e termina num lugar pra fazer determinada tarefa essa ideia de pensar o todo em parte e a parte do todo isso inclusive, nos ajuda a pensar estratégias dentro do fazer da programação, quando a gente pensa em um programa, a gente sabe que tipo, em algum momento a gente vai trabalhar as partes isoladas depois pensar as partes juntas e próprio assim no algoritmo também e eu acho que certo modo as intervenção já apontava pra esse lugar do fazer, assim da prática, apesar



de terem sido aulas mais teóricas. [Gabriel] É, eu vou no mesmo caminho, agora eu diria que ser útil, é mais uma questão mesmo de você tomar consciência de coisas que você tá fazendo no dia-a-dia, de que a gente já faz, só que a gente talvez não fosse consciente dessas coisas de composição, decomposição, as próprias abstrações, justamente por que, a gente não é dessa área, então a gente faz essas atividades em nosso dia-a-dia, que já levam em conta essas coisas, mas com as intervenções a gente meio que “ó isso aqui ó, tá presente nisso aqui, isso que você faz” e tal, e ao mesmo tempo é útil justamente por que, meio que ajuda a gente enxergar a programação ali além do código, no sentido de não é apenas digitar o código, não isso aqui envolve essas mesmas atividades do seu dia-a-dia e que você costuma fazer para outras coisas então meio que fica mais fácil, você conseguir entender como programar. [Igor] É, mas ajuda a pensar em estratégia mesmo, assim, por exemplo, a questão do perceber os padrões, né? Com isso, é super prático quando a gente vai aplicar, a gente entende por exemplo, quando a gente faz um quiz, e ontem a gente fez um exercício e o outro a gente sabe que existe um padrão, ainda que algumas coisas mudem, coisas se repetem e ai aquilo que aparece na teoria e na prática fica muito evidente, quando a gente coloca a mão na massa mesmo, a mão no código.

[Gabriel] Teve muita relevância a própria coisa da programação e da linguagem, é uma abstração. Você tem uma lógica de programação e você traduzir aquilo para uma linguagem é uma abstração tá passando aquilo ali, entende? Porque, pra mim é uma coisa que é muito óbvia, não sei, no sentido de não tem como não ter ali, entende? São processos que tão na coisa da gente estudar, aprender e manipular as coisas que é isso, é no sentido de você tomar consciência das coisas que já estão presentes ali, entende? O próprio programa é uma abstração, entende? Você quer ter uma determinada ação, tem uma determinada coisa para fazer e ai você coloca o programa pra executar, entende? Pelo menos eu vejo assim, né? Não vejo assim, como uma coisa meio, desprendida assim, não acho que uma coisa incorporada ai no próprio processo. Tipo, eu acho que o próprio fato de a gente não falar muito de abstração já demonstra, como ela, está inerente ali ao processo

[Igor]É, eu acho isso, acho que ela, se for pra comparar, assim por que inclusive a gente não falou que eu acho que ela, é...Não tá tanto nesse lugar micro do fazer isso, assim eu acho que ela está nesse lugar mais amplo.

[Gabriel] Não, esse processo de inventar, é abstração. Não tem muito como fazer essa separação assim. Pelo menos eu acho.

Pesquisador: E a ordem que elas foram apresentadas, foi ideal, a partir do conceito que vocês tinham e tem hoje de programação? [Houve uma explicação das 4 partes que foram apresentadas, porém não está totalmente entendível]

[Bernardo] Assim, eu acho que pra nós, acho que o algoritmo que chegou antes, assim por que o [palavra não entendível] foram outras coisas, foram as intervenções, inclusive assim, muito antes, pelo menos pra mim, quando eu fiz a seleção não lembro se estava esse nome algoritmo assim, mas de certo modo eu já reconheci ali então pra mim a ordem não é essa, que o algoritmo veio antes, e eu acho importante começar com algoritmo porque, eu acho que o algoritmo diz sobre todo o processo né? Do começo ao fim, você vê, quais são as etapas, que uma etapa depende da outra.

[Gabriel] E talvez por ser a coisa mais concreta, né? Concreto assim tipo, você entra em contato com uma coisa, [não ficou entendível no final]

[Igor] [o início não está entendível] E na sequência, ele vai entender o que você está fazendo, acho que também é isso, até mesmo para entender mais sobre é...abstração.

Pesquisador: E vocês utilizaram essas habilidades no trabalho final de vocês?

[Bernardo] Eu acho que sim! Certamente! É!

[Igor] Eu acho que inclusive, acho que como você falou é inerente assim você fala de programação, aliás não profundo conhecedor de programação, mas a programação que conheci tem que ter o algoritmo, tem os padrões que se repetem, ainda enquanto linguagem, uma linguagem tem aqueles padrões de linguagem, de forma que os processos aconteçam, é, que essa composição, decomposição, de você pensar nas partes, eu acho que, acho que é isso, não tem como pensar nisso, sem pensar em todas essas questões, ainda que...Eu acho que seria plenamente possível é...Ter feito o curso e não saber o nome dessas coisas, ainda que elas ajudem, eu acho que elas são surreais. Bem nesse processo assim, de nomear as coisas, pra eu poder chegar pra Bernardo e falar “Ah! Por que meu algoritmo, deu algum problema em alguma etapa”, mas de repente, se eu não tivesse esse nome, eu poderia falar outra palavra, entendeu? Como: “Deu algum problema nas etapas do meu programa”, acho que ajuda, por ser uma linguagem comum pra quem faz programação, assim os instrutores e tal, ajuda na comunicação, para a gente ter uma linguagem mais comum.

[Gabriel] Também concordo, acho que é uma coisa mais útil no sentido mais teórico de você compreender a coisa, do que uma coisa, que tenha uma utilidade prática imediata, é possível uma pessoa fazer um algoritmo sem saber ou dominar conceitualmente o que é

composição, decomposição... [Trecho não entendível][Houve uma explicação complementar sobre o uso dos conceitos vs prática]

Pesquisador: [Fez uma pergunta que não foi possível entender]

[Gabriel] É isso, tem várias, agora ao mesmo tempo [não entendível], por que eu acho que habilidade é exatamente a própria coisa de você fazer o algoritmo, como eu disse, eu acho que isso é um processo que tá subjacente a própria coisa de você programar, porém a diferença é: uma coisa é o indivíduo está consciente de que existem as competências a gente pode dizer assim e outra coisa é você, como é que se diz, e você já programa consciente dessas competências específicas tipo: “olha, eu aqui que eu preciso fazer estrutura de repetição”, e não, eu acho que a própria pessoa quando tá aprendendo a programar se for ensinado a ela estrutura de repetição, tipo, ela não necessariamente vai pensar: “nossa tá aqui a questão de conhecimento padrão” ou então a coisa de você como aconteceu várias vezes você, como aconteceu várias vezes aqui que a gente tinha um problema num programa e aí, Docente 02 vinha junto com a gente pra ajudar e tinha que ir lá em determinadas partes e via que algumas estruturazinhas estavam dando problema e eram coisas muito repetidas, entende? Então tipo, eu não acho que é uma coisa assim, eu acho que tipo, que a única diferença é que quem tem conhecimento nesses conceitos dessas ideias, provavelmente, vai ter um tipo de programação mais consciente, saca?

[Igor] Eu discordo, com essa parte quando você fala que não é uma habilidade, ter conhecimento de padrão, eu acho que é uma habilidade, por que tem gente que não consegue ter conhecimento de padrão, portanto é possível alguém não ter essa habilidade. Você não acha possível? Tem gente que não consegue perceber que a forma se repete assim, ela tem dificuldade de identificar essa repetição, em si.

[Gabriel] Eu acho que talvez identificar qual é a repetição, em sim, é meio que você conseguiu deslocar, essa repetição do todo e aí você...Por exemplo, uma falha de um programa, eu acho que isso sim seria uma habilidade, agora não sei se reconhecimento de padrões, em si, é um tipo de habilidade, não sei.

[Igor]Mas habilidade pode ser adquirida [Entrevistado] [parte inaudível] a pessoa ter essa habilidade, é algo que facilita. O programar, ele tem que melhorar o processo, então através do esquema de padrão.

[Igor] Eu acho assim também, é fundamental, eu vou dá meu exemplo: eu, Bernardo sabe disso, eu apanhei muito, no scanf, por que eu sempre esquecia do , que é uma repetição, mas assim eu partia de outras repetições, a repetição do print, que não tem o , entendeu?

Então essa repetição se é fundamental, pro programa é funcionar, eu tive que apanhar muitas vezes, para adquirir isso, sim agora não tem um scan, pra não tem um . Entender que aquela coisa, precisa ser sempre da mesmíssima forma para funcionar...

[Gabriel] Eu acho que são coisas diferentes, pelo menos é como enxergo, talvez eu esteja enxergando a coisa muito conceitualmente, por que acho que o reconhecimento de padrões, eu acho que é uma habilidade humana assim, fundamental pra gente na sociedade, em tudo assim interações, a própria questão de linguagem, a própria questão de hábitos, então tipo eu acho que esse tipo de habilidade que você tá falando é uma coisa muito pontual que pra mim que é muito pontual de programação bem pra você conseguir identificar erros, você conseguir identificar assim, existem estruturas que se repetem e a coisa só funciona toda se estrutura se repetir exatamente da mesma forma aqui, entende? E ai você consegue identificar esse erro dessa estrutura o porquê que tá dando problema lá no final e consertar isso, mas tipo isso é algo que depende de reconhecimento de padrão, eu acho que sim, que é uma coisa que depende algumas pessoas vão ter essa habilidade de fazer isso? Eu acho que não. De fato. Agora, é isso, eu fico me perguntando se, esses conceitos tão gerais como reconhecimento de padrão, composição, decomposição, abstração, é isso, não são coisas de fato, muito mais amplas, muito mais macro, entende? E que a gente tem que lidar no dia-a-dia e que em programação elas surgem de forma pontual assim, justamente por que o programa meio que... O computador meio que emula a nossa cabeça, não sei, o funcionamento da mente e a gente precisa meio que cria uma estrutura do mesmo modo que a gente pensa, sei lá, talvez eu esteja viajando mais do que eu deveria viajar. Docente 02 tem a capacidade de chegar, por exemplo, eu e Sergio estamos fazendo um programa e tá tendo um problema e a gente olhando aqui não consegue, já vimos e revimos e voltamos, e nada, e Docente 02 vem, por que Docente 02 tem um conhecimento maior da linguagem e ela consegue reconhecer um aspecto específico ali e diz: “olha o problema é isso aqui, tá escrito errado!”. Ou então foi um dos outros rapazes que estavam aqui no outro dia, lembra da coisa do...Era das aspas e das aspas duplas, entende? Pra ele conseguir pegar o caractere, e ai a gente ficou boiando, se perguntando “Por que que não tá dando certo?”, Dai ele chegou falou: “isso aqui não pode, tem que ter uma aspas só, não podem ser aspas duplas”. Entende? É isso ai, é nesse sentido que eu tô falando, o problema não estava no nosso reconhecimento de padrões, a gente conseguia reconhecer que tinha um problema, conseguia reconhecer os lugares que estavam tendo o problema, mas a gente não sabia. Então, eu tô dizendo que a gente aplicava as coisas, tipo, por exemplo, Docente 02 aplicava as coisas para de bugar, pra dá o resultado antes pra ver e estava saindo o

resultado e estava saindo antes correto, mas a gente porque, não estava sabendo sacar a coisa da linguagem a gente não conseguia fazer a correção, a gente sabia qual era o problema, sabia que ele estava localizado ali naquela região, por que era o padrão que se repetia, por que a gente modificava as outras partes e não mudava nada, e quando modificava lá aí dava o problema, entende? Mas o fato de a gente não ter o conhecimento específico em função da questão da linguagem ali, tem essa questão de termo, mudou tudo e nesse sentido que eu estou falando.

[Igor] Queria dizer que pra mim, isso, eu não concordo. Realmente, eu acho que reconhecimento de padrão é uma habilidade, tem gente que tem uma facilidade muito grande em identificar um padrão e tem que apanhar muitas vezes, mas se... [Gabriel] Como eu disse, talvez eu esteja... Eu não acho que a gente esteja em discordância, eu acho eu seja apenas a questão do ponto de vista, de como eu estou enxergando o conceito de reconhecimento de padrão, num coisa mais geral e mais macro e você já está enxergando ele, como uma habilidade específica, e é uma possibilidade por que é uma palavra que pode nomear duas coisas diferente, tá entendendo?

Pesquisador: [Pergunta não audível]

[Gabriel] Não, você está falando aí de gramática, não é uma questão de padrão, é uma questão de regra, é uma questão de regra...

Pesquisador: Quero saber o seu conceito de reconhecimento de padrão?

[Gabriel] Você não acha a gramática um padrão?

[Gabriel] É, mas é isso que eu ia falar agora, esse conceito de padrão que ele está usando para falar com relação a essa coisa do programa, é uma questão de regra, a regra diz: “pra ele receber o caractere lá, no coisinha, ele deve usar uma aspa e não duas aspas”, saca a diferença, do que eu estou dizendo? Aí a gente já tá usando um outro fundo, uma outra noção de padrão, que é uma noção de padrão enquanto regra, entende? Tipo, que seria a troca de linguagem, em C, a linguagem em C demanda o que? Que pra ele adicionar, e ficar no programa e contar a pontuação lá do contador ele precisa de uma aspa só e a gente estava colocando duas, entende? É isso que estou falando, isso é um padrão enquanto regra, agora o que eu estou imaginando como reconhecimento de padrão é, por exemplo, a gente vive aqui e todo dia amanhece e anoitece é uma repetição, entende? Ou então você ver formatos, você olha uma forma meio opaca, ou sei lá, embaçada a distância né? Você percebe que aquilo tem um formato meio arredondado e uma coisa meio assim reta, você vai deduzir que é o que? Uma árvore, por exemplo, é nesse sentido que estou falando,

reconhecimento de padrão eu acho que é uma habilidade específica da mente humana nesse sentido, a gente olha as coisas do nosso dia-a-dia, a gente vai reconhecendo esses padrões que se repetem e meio que a gente vai alocando isso em noções, entende? Isso eu acho que é um reconhecimento de padrões enquanto conceito geral e o que eu estou dizendo é a habilidade enquanto indivíduo, é como eu falei, eu acho que essas competências que você apresentou pra gente nas intervenções, são coisas assim que estão presentes na nossa vida, no dia-a-dia pra gente fazer várias coisas, mas a gente faz no automático e quando você vai programar são coisas que se você tiver tomado de consciência delas, meio que você consegue programar de forma mais consciente e te ajudar a fazer aquilo ali, entende? Mas eu não acho assim que é no caso, tipo assim é muito possível que alguém programe, sem ter a noção do conceito consciente na cabeça dessas coisas, é isso que eu estou querendo dizer, entende? Tipo, ela vai ter reconhecimento de padrões, agora assim, eu acho que essa questão de reconhecimento de padrões, enquanto habilidade pontual, enquanto uma coisa assim no sentido tipo eu domino a linguagem C, eu consigo perceber assim a necessidade, eu domino a regra eu sei que a regra vai dizer que aquilo ali tem uma aspa é uma outra coisa diferente, entende? Mas eu acho que eu estou viajando mais do que a própria coisa precisa.

**APÊNDICE J – Diário de Bordo****DIÁRIOS DE BORDO**

Data: 05/05/2018 - Turma B

Desenvolvimento: A aula foi um apanhado inicial sobre o que elas aprenderiam ao longo do curso. A princípio o Danilo, fez uma intervenção sobre raciocínio computacional com elas, sem que antes fossem apresentados conceitos básicos quem são elencados logo na primeira aula. Por se tratar de uma turma pequena, não houve uma dinâmica de apresentação, apenas um bate papo sobre quem eram elas, o que faziam e qual contato já tinha com a área de TI. Seguindo a aula elas interagiram lembrando de conceitos e pontos apresentados pelo Danilo sobre raciocínio computacional e relacionaram com os conceitos de lógica de programação. Finalizamos com a explicação e inscrição das meninas no moodle da turma.

Facilidades: por se tratar de uma turma menor, a condução da aula e o contato mostrou-se mais eficaz

Dificuldades: algumas máquinas apresentaram problemas e reiniciaram ou não ligavam.

Data: 12/05/2018 - Turma B

Desenvolvimento: Nesta aula já foram introduzidos aspectos de lógica e da linguagem C. Algumas sentiram dificuldades em entender as características da linguagem, então fizemos alguns exercícios sobre a sintaxe da linguagem para que acompanhassem o que estava acontecendo no código de exemplo que foi apresentado no slide. Por conta da nova intervenção feita por Danilo, contamos com menos tempo para concluir os exercícios que planejei na aula, então adiei. Ficou combinado de reservarmos um tempo da aula para resolver os exercícios do moodle, até como uma forma de ver o progresso delas e dificuldades mais de perto.

Facilidades: nada a relatar.

Dificuldades: falta de quadro/lousa no espaço para fazer as exemplificações

Data: 15/05/2018 - Turma A

Desenvolvimento: Intervenção de Danilo, onde ele ensinou decomposição e passou uma atividade. Depois disso, Docente 02 iniciou a aula fazendo uma breve revisão da aula anterior, e ensinou um pouco sobre estrutura condicional/seleção.

Facilidades: Os alunos atenderam a intervenção de Danilo, fizeram a atividade sem dificuldade, e compreenderam o assunto.

Dificuldades: A intervenção de Danilo foi um pouco longa, e Docente 02 teve 30 minutos para explicar o aula.

Data: 22/05/2018 - Turma A

Desenvolvimento: Os alunos fizeram exercícios sobre estruturas de repetição para fixar o assunto. Alguns tiveram mais dificuldade que outros, Augusto nosso aluno tem muita dificuldade em 'criar' os algoritmos e acaba pescando de quem está do lado, mas ele se empenha em entender o que está sendo feito.

Facilidades: Os computadores funcionaram normalmente, os alunos se empenharam para resolver os exercícios.

Dificuldades: Alguns alunos tiveram dificuldade em resolver os exercícios, mas resolveram.

Data: 05/06/2018 - Turma A

Desenvolvimento: Docente 02 iniciou a aula com uma revisão sobre os operadores lógicos, pois na aula desse assunto houve muito barulho externo, o que dificultou a compreensão pelos alunos. E como Igor tinha faltado essa aula, tanto Docente 02 quanto os outros alunos explicaram sobre operadores lógicos para ele. Após isso, ela explicou sobre as estruturas de repetição 'if', 'while', 'for', os alunos compreenderam a teoria. Após a explicação, foi iniciada a parte prática da aula, com exercícios do moodle. Alguns alunos tiveram mais dificuldade que outros.

Facilidades: Os alunos compreenderam a explicação.

Dificuldades: Os alunos tiveram dificuldade na resolução dos exercícios, pois estes começaram a ficar mais complexos.

Data: 07/06/2018 - Turma A

Desenvolvimento: Revisão das estruturas de repetição. Exercícios para fixação do conteúdo, e tirar dúvidas.

Facilidades: A turma é empenhada, e a maioria tem facilidade na compreensão do assunto.



Dificuldades: Um aluno tem dificuldade em assimilar o conteúdo, em fazer os exercícios, e acaba pescando a resposta de algum colega. Mas, explicamos e ele compreende o que está sendo feito.

Data: 09/06/2018 - Turma B

Desenvolvimento: Começamos nossa atividade do última aula com a intervenção do aluno Danilo, o mesmo passou alguns conceitos que serão relevantes para construção do seu projeto de conclusão de curso. Danilo fez uma breve revisão de conceitos como algoritmo, sequência lógica e decomposição de problemas. Esta revisão nos ajudou a dar início a aula, pois devido a falta de aulas as meninas precisam relembrar assuntos que já foram passados anteriormente. Após a intervenção, Docente 01 realizou uma revisão para relembrar os conteúdos das aulas passadas onde ela apresentou conceitos básicos, também tocou no assunto sequência lógica, falamos de entrada e saída focando em variáveis e constantes, atribuição de valores nas variáveis e constantes, um pouco sobre boas práticas de programação, processamentos focando em input e outputs como scanf e printf, e operadores aritméticos. Após essa revisão fizemos algumas atividades do bloco dois coletivamente, usando o repl.it. Essas atividades serviram para colocar em prática os conteúdos passados na revisão. Ao final das atividades, Docente 01 apresentou para as meninas o conceitos de operadores relacionais, operadores lógicos, prioridade entre operadores lógicos e uma breve atividade de fixação que havia no slide.

Facilidades: A intervenção feita por Danilo nos ajudou bastante, pois fez com que as meninas relembassem de alguns conceitos que já haviam sido apresentados.

Dificuldades: Devido ao longo período sem aula, as meninas estavam desmotivadas. Metade das alunas faltaram e uma desistiu. Durante a aula uma das alunas falou sobre a baixa motivação e comentou sobre a possibilidade de deixar o curso, porém no decorrer da aula, quando ela conseguiu fazer as atividades de fixação ela ficou motivada a continuar.

Data: 12/06/2018 - Turma A

Desenvolvimento: Docente 02 e Monitor 02 explicaram sobre vetores, apresentaram a plataforma URI, e aplicaram alguns exercícios para fixar o assunto de vetores.

Facilidades: Os alunos compreenderam o assunto com facilidade.

Dificuldades: Os alunos enfrentaram dificuldade em realizar os exercícios, mas conseguiram resolver os problemas.

Data: 14/06/2018 - Turma A

Desenvolvimento: Bianca iniciou a aula com aplicação de um questionário físico, Danilo fez uma checagem” dos alunos que participaram das intervenções dele, e marcou um horário com os que não participaram da intervenção. Depois, Bianca aplicou outros questionários online para os alunos, e explicou sobre a gamificação. E às 19:30 Docente 02 iniciou propriamente a aula sobre switch/case”. Os alunos compreenderam o assunto, e houve a parte prática da aula, aplicamos exercícios do URI.

Facilidades: Os alunos compreenderam o assunto, e se empenharam em resolver e entender o assunto.

Dificuldades: A questão do tempo, muitas atividades foram aplicadas, alguns alunos se cansaram até de responder tanto questionário.

Data: 16/06/2018 - Turma B

Desenvolvimento: Começamos a aula resolvendo exercícios em conjunto, e após revermos o conceito de tipagem de variáveis. Em seguida vimos o novo conteúdo, estruturas de controle condicional ou de seleção, e para fixar o conhecimento fizemos exercícios em conjunto novamente. Os exercícios aplicados em

Facilidades: Os exercícios aplicados em sala tem feito as meninas evoluir com mais rapidez no conteúdo, e gerando dúvidas pertinentes que nós conseguimos sanar na hora. Nossas alunas participam bastante da aula, em especial Aurora, essa aluna é a que mais questiona em sala e sempre tem curiosidades que vão além do conteúdo.

Dificuldades: Novamente só tivemos 3 alunas em sala, as outras 4 faltaram. E uma das aulas acabou chegando com uma hora e vinte de atraso. Essa mesma aluna não tinha vindo na aula anterior, e Docente 01 teve que sentar com ela para fazer o exercício juntas e ir explicando o conteúdo que ela pediu.