VIDYA

VIDYA, v. 44, n. 2, p. 153-173, jul./dez. - Santa Maria, 2024.

Recebido em: 06.06.2024. Aprovado em: 17.08.2024.
ISSN 2176-4603 | DOI: doi.org/10.37781/vidya.v44i2.4989

EXPLORANDO O PENSAMENTO COMPUTACIONAL COM A RECURSÃO NA FORMAÇÃO INICIAL DE PROFESSORES DE MATEMÁTICA: REFLEXÕES A PARTIR DE ROBBIE CASE

EXPLORING COMPUTATIONAL THINKING WITH RECURSION: REFLECTIONS FROM ROBBIE CASE

EXPLORANDO EL PENSAMIENTO COMPUTACIONAL CON RECURSIVIDAD: REFLEXIONES SEGÚN ROBBIE CASE

ANA PAULA CANAL[1]
VANILDE BISOGNIN[2]
SILVIA MARIA DE AGUIAR ISAIA[3]

RESUMO

Este artigo traz reflexões sobre o Pensamento Computacional trabalhado com a recursão, a partir de Robbie Case. A investigação procurou responder à pergunta: como o Pensamento Computacional, no ensino, pode contribuir para a formação do futuro professor de Matemática? A pesquisa caracterizou-se por ser qualitativa, do tipo estudo de caso. O conteúdo abordado, Padrões e Regularidades, integrou-se ao Pensamento Computacional, com a recursão na Linguagem Python. As fontes de evidência foram questionário, observação participante, diário de campo, artefato físico/cultural e entrevistas. A análise dos dados foi baseada na teoria de Robbie Case. Os resultados mostram que os licenciandos desenvolveram habilidades do Pensamento Computacional e conseguiram propor e resolver problemas com a recursão. No estudo de caso, foram promovidas relações entre a generalização (Pensamento Algébrico) e as habilidades de abstração, coleção, análise, representação de dados e algoritmos (Pensamento Computacional).

Palavras-chave: Computação na Educação; Pensamento Algébrico; Padrões e Regularidades; Linguagem Python.

ABSTRACT

This article presents reflections on Computational Thinking applied through recursion, based on the theories of Robbie Case. The investigation sought to answer the question: how can Computational Thinking, in teaching, contribute to the training of future Mathematics teachers? The research was characterized as qualitative and of the case study type. The content covered, Patterns and Regularities, was integrated with Computational Thinking, specifically through recursion in the Python Language. Evidence sources included a questionnaire, participant observation, field diary, physical/cultural artifact, and interviews. The data analysis was developed using Robbie Case's theory. The results show that the students developed Computational Thinking skills and were able to propose and solve problems using recursion. In the case study, relationships were established between generalization (Algebraic Thinking) and the skills of abstraction, collection, analysis, data representation, and algorithms (Computational Thinking).

Keywords: *Computing in Education; Algebraic Thinking; Patterns and Regularities; Python Language.*

1 Doutora em Ensino de Ciências e Matemática. Universidade Franciscana. E-mail: apc@ufn.edu.br. ORCID: https://orcid.org/0000-0001-6360-1688
2 Doutora em Matemática. Universidade Franciscana. E-mail: vanilde@ufn.edu.br. ORCID: https://orcid.org/0000-0001-5718-4777
3 Doutora em Educação. Universidade Franciscana. E-mail: sisaia@ufn.edu.br. ORCID: https://orcid.org/0000-0002-9987-7931

RESUMEN

Este artículo presenta reflexiones sobre el Pensamiento Computacional aplicado a través de la recursividad, basadas en las teorías de Robbie Case. La investigación buscó responder a la pregunta: ¿cómo puede el Pensamiento Computacional, en la enseñanza, contribuir a la formación de futuros profesores de Matemáticas? La investigación se caracterizó como cualitativa y del tipo estudio de caso. El contenido cubierto, Patrones y Regularidades, se integró con el Pensamiento Computacional, específicamente a través de la recursividad en el Lenguaje Python. Las fuentes de evidencia fueron cuestionario, observación participante, diario de campo, artefacto físico/cultural y entrevistas. El análisis de los datos se elaboró con la teoría de Robbie Case. Los resultados muestran que los estudiantes desarrollaron habilidades de Pensamiento Computacional y fueron capaces de proponer y resolver problemas utilizando recursividad. En el estudio de caso, se establecieron relaciones entre la generalización (Pensamiento Algebraico) y las habilidades de abstracción, recopilación, análisis, representación de datos y algoritmos (Pensamiento Computacional).

Palabras-clave: *Computación en la Educación; Pensamiento Algebraico; Patrones y Regularidades; Lenguaje Python.*

INTRODUÇÃO

O ensino das diferentes áreas do conhecimento está imerso nas transformações que perpassam a sociedade e reflete essas transformações. Uma das habilidades necessárias para os cidadãos atuarem na sociedade, em constante transformação, é o Pensamento Computacional.

O Pensamento Computacional é uma forma de solucionar problemas de diferentes áreas baseando-se em conceitos da Ciência da Computação (WING, 2006). São empregadas ferramentas mentais que envolvem um pensamento com múltiplos níveis de abstração. Uma forma de desenvolver o Pensamento Computacional é incluí-lo nas diferentes etapas de ensino.

Nos últimos anos, discussões foram promovidas a respeito da inclusão da Computação na Educação Básica, no Brasil. Formalmente, culminou com as Normas sobre Computação na Educação Básica, Complemento à Base Nacional Comum Curricular (BNCC), Parecer CNE/CEB nº 2/2022 de 17 de fevereiro de 2022. Esse parecer foi elaborado em conjunto com professores e pesquisadores de áreas como Pedagogia, Geografia, Matemática, Física, Biologia, Educação Física, Comunicação Social, e Computação, que atuam em diferentes etapas de ensino. Houve também a colaboração com organizações e universidades internacionais (BRASIL, 2022a). Portanto, há necessidade do trabalho junto aos professores e futuros professores para se capacitarem e se tornarem atuantes neste processo.

Esse artigo explora o Pensamento Computacional com a recursão. A recursão é compreendida como a definição de um objeto, função, sequência ou conjunto a partir de si próprio (NCTM, 2000). É um conceito presente na Matemática e na Computação. O artigo traz reflexões, a partir do neopiagetiano Robbie Case, sobre esta pesquisa em que o Pensamento Computacional foi incluído no ensino de forma transversal à Matemática, com o conteúdo Padrões e Regularidades.

Foi realizado um estudo de caso na formação inicial de professores de Matemática, em busca da resposta à pergunta: Como o Pensamento Computacional, no ensino, pode contribuir para a formação do futuro professor de Matemática?

Os processos de ensinar e de aprender precisam estar apoiados em teorias de aprendizagem visando ao desenvolvimento cognitivo do aprendiz. A teoria de Robbie Case (1989) considera que o desenvolvimento das estruturas cognitivas é promovido pela resolução de problemas. Para Case, a resolução de problemas é amparada pelo contexto da resolução, o meio em que o indivíduo está se desenvolvendo, as interações nesse meio e a cultura. Como recurso para explorar o Pensamento Computacional, a linguagem de programação Python foi utilizada.

A fundamentação teórica deste trabalho traz conceitos de Padrões e Regularidades, Pensamento Computacional, Recursão e da teoria de aprendizagem de Robbie Case. Nas próximas seções do artigo, esses aspectos são caracterizados.

FUNDAMENTAÇÃO TEÓRICA

Padrões correspondem à ordem e possibilitam a identificação da ordem e a organização de diferentes aspectos do mundo. Os padrões permeiam o dia-a-dia (PEREIRA, 2004). As crianças ao classificar e ordenar objetos, podem estar fazendo a descoberta de padrões, é uma descoberta que pode acontecer desde cedo.

"Um padrão será de repetição quando há um motivo identificável que se repete de forma cíclica indefinidamente. Um padrão será de crescimento quando cada termo muda de forma previsível em relação ao anterior." (VALE, 2012, p.186). A ideia fundamental num padrão envolve repetição e crescimento (mudança).

Com Padrões e Regularidades, a generalização pode ser trabalhada constantemente. Generalizar é o processo pelo qual, a partir de um conjunto de casos particulares, o raciocínio continua além deste conjunto de casos, identificando a regularidade entre eles, fazendo a generalização da ideia por meio do discurso e/ou da expressão formal (BLANTON; KAPUT, 2005).

O desenvolvimento do raciocínio a partir das generalizações, pode levar ao pensamento recursivo. Entende-se que a definição recursiva é aquela que define um objeto (função, conjunto...) em termos de si próprio. A procura de padrões e regularidades e a formulação de generalizações em diversas situações devem ser fomentadas desde os primeiros anos da educação básica (NCTM, 2000).

A compreensão dos padrões é recomendada por meio da sua descrição, da generalização, da representação dos padrões em diferentes formas: geométrica, numérica, simbólica. A representação de padrões deve ser trabalhada, conforme a etapa de ensino e assim, quando possível, definir explicitamente os padrões, até chegar à definição recursiva (NCTM, 2000).

A definição recursiva é usada para descrever sequências, algoritmos ou outros objetos de forma que cada elemento é definido a partir de termos anteriores. Por exemplo, uma função é recursiva, se faz invocação a si mesma diretamente ou chama outra função que faz a invocação à função original.

Na definição recursiva, há um caso base para a parada da recursão. Algebricamente, pode-se definir uma função f_n como recursiva, se existe uma sequência de funções $f_i \dots f_n$, em que f_i, para $1 \leq i \leq n$, é igual ao caso base ou f_i é definida por meio da função f (ACZEL, 1977). A recursão é um princípio que pode ser utilizado para resolver problemas, os quais podem ser decompostos em problemas menores, similares ao problema original.

O raciocínio algébrico é uma forma de pensar que supõe o estabelecimento de generalizações em diversas situações matemáticas. "Este tipo de raciocínio está no coração da matemática concebida como a ciência dos padrões." (GODINO; FONT, 2003, p.774). Nesse sentido, vai ao encontro do Pensamento Computacional, pois conforme Wing (2011), esse pensamento envolve o reconhecimento de padrões, algoritmos, paralelismo, raciocínio para decompor problemas e o pensamento recursivo.

Assim como na Matemática, na Computação a recursão está presente. Algoritmos recursivos dividem o problema a ser resolvido, em problemas menores ou mais simples, resolvem os problemas menores e combinam as soluções até chegar à solução do problema original (CORMEN *et al.*, 2001).

Um algoritmo recursivo invoca a si mesmo e possui um caso base, em pelo menos um local do algoritmo. Para o desenvolvimento de algoritmos recursivos, são utilizadas funções (ou métodos) que fazem uso de recursão (contém pelo menos uma invocação a si próprio).

Para Ribeiro et al (2019), o Pensamento Computacional refere-se à habilidade de compreender, analisar, definir, modelar, resolver, comparar e automatizar problemas,, por meio da criação de algoritmos, da aplicação de fundamentos da computação para aprimorar a aprendizagem e o pensamento criativo nas diversas áreas do conhecimento.

Acredita-se na integração do Pensamento Computacional com as outras ciências. Neste estudo de caso, com a Matemática, no ensino para a formação inicial de professores de Matemática, por meio dos Padrões e Regularidades.

Os cursos de licenciatura em Matemática têm como objetivo formar professores para a Educação Básica. O desenvolvimento do raciocínio lógico, da capacidade de resolução de problemas, de fazer uso de novas linguagens e tecnologias, do trabalho colaborativo e interdisciplinar, contribuem para a atuação do futuro professor de Matemática.

Ao licenciando, é necessário conhecimentos específicos de sua área, conseguir mobilizá-los, transformando-os em ação para atender demandas de sua profissão. A capacidade de articulá-los, relacionando à realidade e a diferentes situações vivenciadas na sala de aula, se constrói no decorrer da sua trajetória de formação e atuação.

Na formação inicial, é importante promover práticas, o trabalho integrado e vivências múltiplas. Aquilo que os licenciandos experimentam e é positivo, pode ser incorporado em suas práticas pedagógicas futuras. Os cursos de formação de professores em áreas específicas, como a Matemática, devem proporcionar uma formação, trabalhando a ideia de que as diferentes áreas do conhecimento se interconectam, para a resolução de situações reais.

Os cursos de licenciatura em Matemática devem ser organizados de forma a desenvolver Competências e Habilidades, dentre estas, destacam-se aquelas diretamente relacionadas ao presente trabalho, como: “c) capacidade de compreender, criticar e utilizar novas ideias e tecnologias para a resolução de problemas. [...] f) estabelecer relações entre a Matemática e outras áreas do conhecimento” (BRASIL, 2002, p.4).

A Resolução CEB 01/2022 de 04 de outubro de 2022, Normas sobre Computação na Educação Básica - Complemento à BNCC, define que cabe aos Estados, aos Municípios e ao Distrito Federal iniciar a implementação desta diretriz até um ano após homologação (BRASIL, 2022b). Neste contexto, é imprescindível trabalhar o Pensamento Computacional, na formação de professores.

Entende-se que a inserção do Pensamento Computacional no ensino para a formação de professores de Matemática, pode promover o desenvolvimento das competências/habilidades salientadas.

Instigar o Pensamento Computacional com padrões e regularidades é uma possibilidade para a inclusão da Computação na Educação Básica. Neste estudo, como recurso para trabalhar o Pensamento Computacional, a linguagem Python foi utilizada, por ser de alto nível, aproximar-se da linguagem natural e, com isso, muitos detalhes da arquitetura do computador não precisam ser considerados para o seu uso, proporcionando maior abstração. Outros motivos foram, por Python ser uma linguagem de código aberto: o seu direito autoral permite a distribuição para qualquer pessoa e para qualquer finalidade e por ser considerada uma linguagem de programação de fácil aprendizagem (ROMANO, 2015).

Esta pesquisa é um estudo de caso e foi amparada pela Teoria de Robbie Case. Case investigou e desenvolveu a teoria, cujo enfoque principal é a Estrutura de Controle Executivo (ECE), estrutura que contempla as estratégias empregadas pelos indivíduos na resolução de problemas.

VIDYA, v. 44, n. 2, p. 153-173, jul./dez. - Santa Maria, 2024.

A aplicação de diversas estruturas de controle executivo proporcionam a criação de uma estrutura de ordem superior, promovendo o desenvolvimento do indivíduo. Os indivíduos são dotados de processos reguladores, desde cedo. Esses processos diferem entre si, se são impulsionados pelo objetivo e pela facilitação social. Por meio dos processos reguladores, novas estruturas de controle executivo são elaboradas (CASE, 1989).

Os processos reguladores, segundo Case (1989) são: Resolução de Problemas, Exploração, Imitação e Regulação Mútua. Esses dois últimos processos, são mediados pela interação humana.

O processo regulador chamado Resolução de Problemas é aquele em que o sujeito diante de um objetivo que não pode ser alcançado imediatamente por uma sequência operacional existente, procura uma nova sequência operacional a fim de alcançá-lo (CASE, 1989).

O processo regulador Exploração é impulsionado pela situação, uma estratégia já conhecida pode ser aplicada a uma situação, mas não se pode prever os resultados. O processo regulador Imitação consiste na imitação da ação ou conduta, por parte do indivíduo com estrutura de controle executivo de ordem inferior, ao observar a ação ou conduta de outro indivíduo mais experiente (CASE, 1989).

O processo de Regulação Mútua envolve a adaptação ativa do indivíduo e de outro ser humano aos sentimentos, cognições ou condutas de cada um (CASE, 1989). Cada membro do grupo exerce uma influência mútua sobre o outro.

Em um segundo momento de sua investigação, Case (1992) identificou que junto às Estruturas de Controle Executivo, para cada domínio do conhecimento, havia algo mais amplo. Ele a nomeou como Estrutura Conceitual Central (ECC), a estrutura que ampara a definição e execução de estratégias para solução de problemas, pelo indivíduo, inserido em sua cultura.

A Estrutura Conceitual Central contempla as relações entre os significados envolvidos na resolução de problemas, as relações semânticas entre os diferentes nodos (nós) que representam os conceitos gerais e suas representações (CASE, 1992).

Na teoria de Robbie Case, destacam-se os seguintes aspectos: os indivíduos, no processo de aprendizagem, fazem uso do conhecimento que já possuem (as estruturas já construídas) e desenvolvem novas estruturas para resolver problemas e buscar objetivos conforme seus interesses, no ambiente em que estão inseridos, e por isso, são ativos neste processo de desenvolvimento. O desenvolvimento do indivíduo pode variar de um domínio ao outro, pois depende da oportunidade que ele tem em explorar o domínio e resolver os problemas que lhe são apresentados.

O ensino é aspecto essencial, segundo Case, para proporcionar o desenvolvimento do indivíduo. Por meio do ensino, acontecem os processos reguladores gerais, conforme a cultura e consequentemente a criação e ampliação das estruturas executivas e conceituais. O foco é o ensino na formação inicial de professores, com o trabalho do Pensamento Computacional, Padrões e Regularidades e a recursão.

METODOLOGIA

A abordagem da pesquisa é qualitativa, caracterizando-se por ser geral e ampla, voltada à descrição e entendimento a partir da perspectiva dos sujeitos e sua interpretação. O tipo desta pesquisa é Estudo de Caso, que é mais apropriado, em comparação a outros, quando existem três situações:

> (1) as principais questões da pesquisa são "como?" ou "por quê?"; (2) um pesquisador tem pouco ou nenhum controle sobre eventos comportamentais; e (3) o foco de estudo é um fenômeno contemporâneo, em vez de um fenômeno completamente histórico. (YIN, 2015, p.2).

O estudo de caso, nesta pesquisa, parte do desejo da compreensão detalhada de: um pequeno número de sujeitos. A unidade de análise foi composta pelo grupo de quatro licenciandos em Matemática da Universidade XXX, que cursaram a disciplina "Pensamento Computacional e o Ensino de Matemática: uma abordagem sobre padrões", de quarenta e quatro horas, oferecida como Atividade Curricular Complementar.

Para esta pesquisa:

> "O uso de múltiplas fontes de evidência propicia profundidade do estudo, a identificação de aspectos que talvez, por uma única fonte de evidência, estariam ocultos. Também, favorece a análise dos dados pois as diferentes procedências podem apresentar aspectos convergentes" (YIN, 2015, p.109).

As fontes de evidência nesta pesquisa compuseram-se por questionário, observação participante, diário de campo, artefatos produzidos pelos participantes e entrevistas. Os artefatos são os produtos finais da resolução de problemas pelos sujeitos, como por exemplo, o arquivo que contém um programa na linguagem de programação Python ou a própria resolução do problema desenvolvida com papel e lápis.

A analise e interpretação dos dados foram realizadas a partir dos pressupostos da teoria de Robbie Case. A Estrutura de Controle Executivo definida por Case (1989) é um dos elementos centrais para mapear como os licenciandos estabeleceram estratégias para a resolução dos problemas propostos nas atividades e nos problemas criados por eles.

Outro constructo importante da teoria de Case, para a análise, é a Estrutura Conceitual Central (CASE, 1992). Essa estrutura possibilitou mapear as relações entre os significados envolvidos na resolução de problemas, as relações semânticas entre os diferentes nodos que representam os conceitos gerais e suas representações.

Neste artigo, um recorte da pesquisa é apresentado e discutido no que tange ao trabalho de Padrões e Regularidades, por meio do Pensamento Computacional e o uso da recursão, na Matemática e na Computação. Para trabalhar as habilidades do Pensamento Computacional, na formação inicial de professores, a linguagem de programação Python foi utilizada.

O projeto foi submetido à apreciação do Comitê de Ética em Pesquisa (CEP), da Universidade Franciscana, na Plataforma Brasil e o Parecer número 3.565.731 aprovou a presente pesquisa.

REFLEXÕES A PARTIR DO DESENVOLVIMENTO DO ESTUDO DE CASO

O estudo de caso foi feito com a disciplina "Pensamento Computacional e o Ensino de Matemática: uma abordagem sobre padrões". Este recorte apresentado e suas reflexões são feitos a partir de dois encontros realizados, cada um com quatro horas.

No encontro em que foram trabalhados a recursão e os padrões, os estudantes se organizaram em duplas. A atividade solicitava a definição matemática recursiva da sequência numérica

VIDYA, v. 44, n. 2, p. 153-173,
jul./dez. - Santa Maria, 2024.

formada pelos números ímpares. A definição do grupo A foi conforme a Figura 1(a) e o grupo B, conforme a Figura 1(b).

Figura 1 - Definição matemática da sequência.

$$an = a_1 + (n-1) . 2$$

(a) Equação definida pelo grupo A

$$f(n), \begin{cases} 1, \text{ se } n = 0 \\ 2n, f(2n-1), \text{ se } n \neq 0 \end{cases}$$

(b) Equação definida pelo grupo B

Fonte: Dados da Pesquisa.

O grupo A comentou que chegou à equação apresentada "*Pela definição de progressão aritmética de números ímpares em sequência*." Pode-se dizer que assemelha-se à equação do termo geral e não contemplou a recursão. O grupo B mostrou que para fazer a definição recursiva, começou pelo problema maior e procurou ir reduzindo até chegar ao primeiro termo da sequência e em sua resposta comentaram "*Pensando o contrário da progressiva*."

A definição recursiva poderia ter sido a equação de diferenças, $a_n = a_{n-1}+2$, para $n \geq 1$, sendo $a_1 = 1$. Talvez, pela simplicidade da sequência, chegar a esta equação não foi de forma direta aos estudantes.

O grupo B procurou definir a produção da sequência toda e não a definição de apenas um dos termos. Por isso, para $n \neq 0$, definiram a produção do n, seguido de $f(2n-1)$, ou seja, o elemento n e novamente a chamada à função. O argumento passado à função não produz os elementos adequadamente conforme a sequência, a partir de um número ímpar informado, por exemplo, se $n = 5$ então $5, f(2\times5-1)$ que equivale à $5, f(9)$. Continuando, tem-se: $5, 9, f(2\times9-1)$ que resulta em $5, 9, f(17)$ o que não corresponde à sequência dos números ímpares.

Para a produção de todos os elementos da sequência, poderia ter sido usada a seguinte definição: $f(n) = f(n-1); 2n-1$, se $n>1$, sendo $f(1) = 1$, então se tem a sequência de números ímpares.

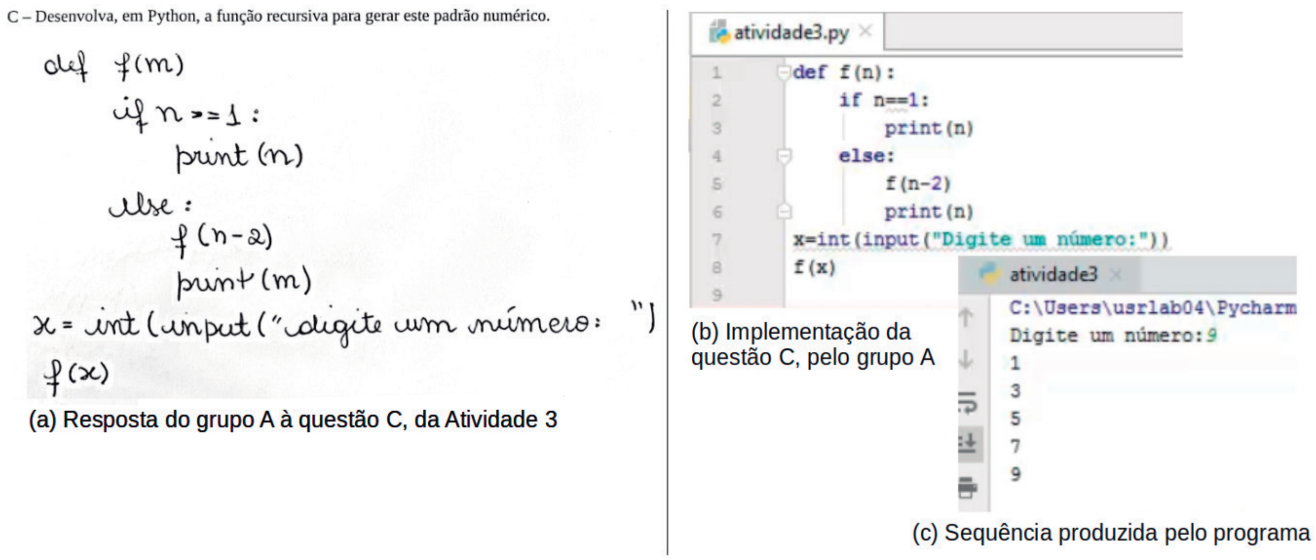
Na continuidade da atividade, foi solicitada a produção do padrão numérico, de forma recursiva, na linguagem Python. O grupo A, inicialmente, discutiu sobre este desenvolvimento, afirmaram que é com o uso de funções e começaram a esboçar o código, com uma função para produzir a sequência, considerando a recursão. O processo de exploração aconteceu no código em Python, pois foram experimentando e validando o resultado. Durante esta resolução, começaram a perceber que deveriam considerar *"a diferença entre os termos"* e comentaram *"Agora está indo para frente."*, e continuaram o raciocínio, indo do termo informado *"até chegar no zero."*, como afirmaram.

VIDYA, v. 44, n. 2, p. 153-173,
jul./dez. - Santa Maria, 2024.

A solução desenvolvida pelo grupo A e respondida na folha da questão, está na Figura 2(a) e a implementação em Python, correspondente está na Figura 2(b). O resultado da execução do programa, a sequência de números ímpares, está no item (c), da mesma Figura.

Essa solução implementou a recursão por meio da função *f(x)*, em que a variável *x* recebe o valor do número digitado pelo usuário e é feita a chamada da função, passando o valor de *x* para o *n*, na função *f*. O caso base é para $n = 1$ e a recursão é implementada no "else", em que a própria função *f* é chamada, tendo como argumento *n-2*, ou seja, *f(n-2)*.

Figura 2 - Implementação da produção recursiva da sequência de números ímpares - Grupo A.



(a) Resposta do grupo A à questão C, da Atividade 3

(b) Implementação da questão C, pelo grupo A

(c) Sequência produzida pelo programa

Fonte: Dados da Pesquisa.

Observa-se que a estratégia relaciona a recursão à implementação das funções recursivas. A definição matemática desenvolvida inicialmente na atividade, não refletiu suficientemente a solução implementada pelo grupo A. Portanto, o desenvolvimento do algoritmo deu-se de forma independente à definição matemática. Além disso, no programa, se o número informado fosse um número par, a sequência produzida seria dos números pares, porém o grupo não constatou esta situação.

O Grupo B, mesmo tendo realizado uma definição recursiva à sequência dos números ímpares, como na Figura 1(b), ao desenvolver o algoritmo não conseguiu transferi-la para a função recursiva, como pode ser observado no código da Figura 3. O grupo produziu essa solução desenvolvida com a estrutura de repetição *for,* sendo a solução iterativa ao problema.

VIDYA, v. 44, n. 2, p. 153-173,
jul./dez. - Santa Maria, 2024.

Figura 3 - Implementação da produção recursiva da sequência de números ímpares - Grupo B.

C – Desenvolva, em Python, a função recursiva para gerar este padrão numérico.

```
def padrao(n):
    if n == 0:
        print("1")
    else:
        i = 1
        for i in range(1, n):
            print(2*i-1)
k = int(input("Valor"))
padrao(K+1)
```

```
def padrao(n):
    if n == 0: ## se for igual a 0 o n, ele recebe 1
        print ("1")
    else:
        i = 1 ## comeca em 1
        for i in range (1,n): ## loop que comeca em 1, e vai ate n
            print(2*i - 1) ## realiza o calculo


k = int(input("Quantos padroes deve ser feito?"))
padrao(k+1) ## como estava mandando valor k-1, para o loop, foi acr
```

Fonte: Dados da Pesquisa.

A solução do Grupo B não contemplou a recursão, uma vez que a função não faz chamada a si mesma. O grupo utilizou outra lógica para o desenvolvimento da solução, trazendo a solução iterativa, com o uso do laço de repetição *for*.

Durante a atividade, foi observado que aconteceram os processos reguladores gerais de Case: Regulação Mútua, Resolução de Problemas e Exploração. A Regulação Mútua ocorreu quando os estudantes interagiram e discutiram, em duplas, sobre a solução. A Resolução de Problemas, quando cada um buscou e propôs individualmente a solução, a partir do que foi discutido ou aprofundando a reflexão e Exploração aconteceu quando os estudantes exploraram o desenvolvimento do algoritmo e sua implementação na linguagem Python, verificando o que era produzido, se poderiam seguir com a solução na implementação do código.

O estabelecimento de relações entre a generalização realizada para a definição recursiva e a elaboração do algoritmo recursivo encontra-se em estágio inicial e precisa ser aprofundado, assim como a via contrária: do algoritmo recursivo à definição matemática recursiva. A limitação destas relações foi evidenciada ao serem observadas as soluções apresentadas pelos grupos. O grupo A, conseguiu a implementação recursiva do algoritmo para produção do padrão, porém a definição matemática não alcançou o esperado. O grupo B, por sua vez, elaborou a definição matemática recursiva, porém a implementação da produção da sequência foi iterativa e, consequentemente, não recursiva.

A coleção e análise de dados, habilidades do Pensamento Computacional, aconteceram ao ser analisada a sequência e buscado identificar o padrão representado.

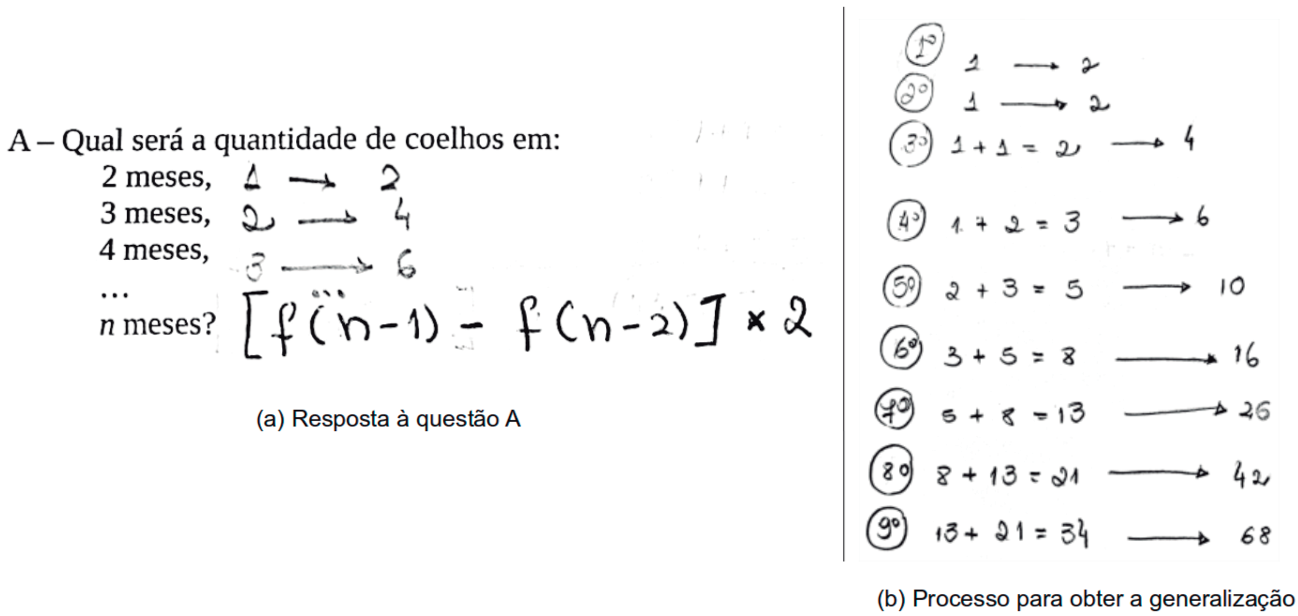
No encontro subsequente, foi desenvolvida outra atividade visando à recursão. Inicialmente, foi apresentado o problema proposto por Fibonacci, mas sem informar a autoria do problema, para que os estudantes investigassem e fizessem suas reflexões. A descrição do problema foi:

> Qual é o número de casais de coelhos ao início de cada mês se um único casal de coelhos recém-nascidos é colocado no interior de um cercado em janeiro e se cada casal gera um novo casal no início do segundo mês após o nascimento e mais um casal no começo de cada mês daí em diante? (GUNDLACH, 1992, p. 62)

A questão "*Qual será a quantidade de coelhos em 2 meses, 3 meses, 4 meses, ..., n meses?*" fez com que, em duplas, houvesse interação entre os estudantes na busca do padrão, considerando os elementos iniciais da sequência. O grupo A, no início, comentou *"contando por mês..."* e assim começaram a identificar a quantidade de casais de coelhos por mês, porém não observaram que no enunciado do problema constava que a reprodução de cada casal de coelhos acontecia a partir do segundo mês de vida, assim que perceberam, começaram a buscar a quantidade por mês e concluíram: *"É a soma dos dois antecessores"* e em seguida exclamaram: *"É a Sequência de Fibonacci!"*.

O processo de verificação por mês da quantidade de casais de coelhos e da quantidade de coelhos, desenvolvido pelo grupo A, está na Figura 4. Pode-se observar que o Grupo A numerou cada mês, identificou a quantidade de casais de coelhos no primeiro e no segundo mês que corresponde a um, pois ainda não havia iniciado a reprodução e, na mesma linha (Figura 4(b)), calcularam a quantidade de coelhos Após o segundo mês, passaram a considerar a reprodução dos coelhos e o consequente aumento do número de casais. Mantiveram em cada linha o número do mês, a quantidade de casais e a quantidade de coelhos.

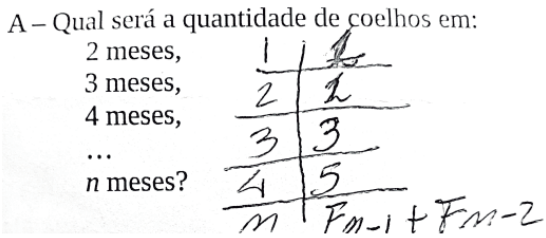
Figura 4 - Processo de generalização na questão A - Grupo A.



Fonte: Dados da Pesquisa.

A partir desses passos, generalizaram a sequência (Figura 4(b)), obtendo a equação que está na Figura 4(a). O grupo ao escrever trocou o operador '+' pelo '-', como pode ser observado, mas a intenção do grupo era a soma, pois comentaram sobre isto, na entrevista e pode ser observado na Figura 4(b). Também, multiplicaram por dois o elemento da sequência, na generalização, pois a pergunta solicitava a quantidade de coelhos e na sequência de Fibonacci, cada elemento representa a quantidade de casais de coelhos. A definição encontrada é a recursiva: cada elemento n da sequência, em que $n \geq 3$ é definido como a soma de seus dois antecessores e representado matematicamente por meio da recursão da função f.

Figura 5 - Processo de generalização na questão A - Grupo B.



Fonte: Dados da Pesquisa.

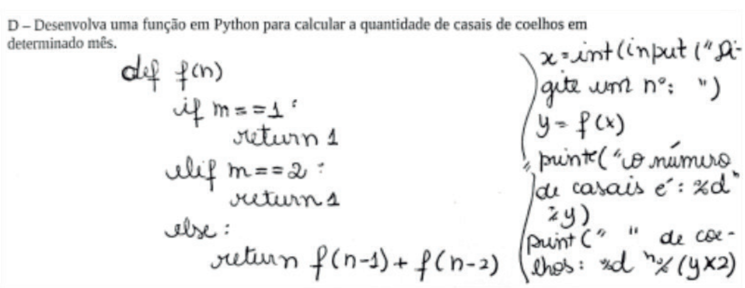
O grupo B elaborou um processo usando uma tabela, em que colocou o número do mês e na segunda coluna, a quantidade de casais de coelhos, como pode ser observado na Figura 5. Ao desenvolverem, comentaram que era *"A fórmula de Matemática Discreta"*, trabalharam para encontrá-la e a relacionaram à Sequência de Fibonacci. Assim, para o mês *n*, definiram a equação com a generalização da sequência, em que está na mesma tabela (Figura 5), na linha do n-ésimo mês.

Percebeu-se que o processo regulador, segundo Case (1989), predominante foi a Regulação Mútua: cada dupla interagiu e discutiu para construir a solução e posteriormente, todos os estudantes interagiram. Ao buscarem estratégias para resolver o novo problema, a partir dos conhecimentos que possuíam caracterizou o processo regulador Resolução de Problemas. O processo Exploração aconteceu durante o desenvolvimento do algoritmo, quando estavam procurando o n-ésimo elemento da sequência, por meio da implementação da generalização definida.

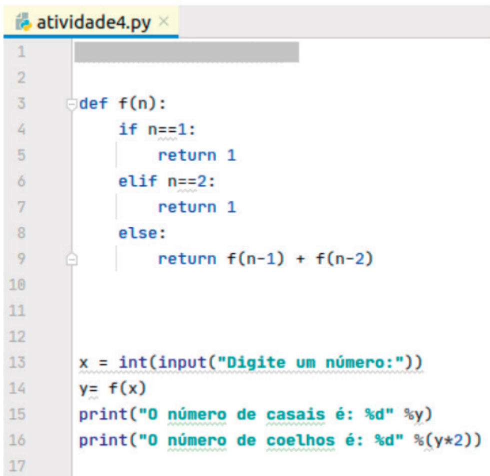
A definição matemática da sequência foi feita pelas duplas e ao serem questionadas: a definição é recursiva?, o grupo A afirmou *"Sim, é recursiva pois depende dos termos anteriores."*, assim como o grupo B.

Na continuidade das atividades, foi solicitado que desenvolvessem uma função em Python para calcular a quantidade de casais em determinado mês. O grupo A elaborou a solução que está na Figura 6.

Figura 6 - Função recursiva para a Sequência de Fibonacci - Grupo A.



(a) Resposta à questão D



(b) Implementação em Python

Fonte: Dados da Pesquisa.

VIDYA, v. 44, n. 2, p. 153-173, jul./dez. - Santa Maria, 2024.

No início da execução do programa, os estudantes solicitaram ao usuário *"Digite um número: "*, que corresponde ao mês para o qual é calculada a quantidade de coelhos. A solução desenvolvida é a recursiva e está completa. A partir dos dois casos base da recursão $f(1) = 1$ e $f(2) = 1$, os demais elementos são calculados recursivamente ao ser invocada a função *f* da mesma forma como na definição matemática: $f(n\text{-}1) + f(n\text{-}2)$, quando $n \geq 3$. O Grupo A teve a preocupação em mostrar ao usuário como resultado o número de casais e o número de coelhos.

O grupo B começou a explorar o desenvolvimento da função em Python de forma iterativa, porém não conseguiram completamente. Observou-se o processo regulador Exploração e a cada tentativa, encontraram diferentes resultados que não se aproximavam daquilo que desejavam. Então, optaram pelo desenvolvimento da função recursiva. Buscaram a definição matemática recursiva elaborada para iniciar o algoritmo. A ideia foi usar a equação desenvolvida, para fazer a chamada recursiva da função, para os casos em que $n \geq 3$ (Figura 7, linhas 11 e 12), e definir como casos base, os valores da função para $n = 1$ e $n = 2$, com o valor *1* (Figura 7, linhas 7 a 10).

Figura 7 - Função recursiva para a Sequência de Fibonacci - Grupo B.

```
ex2.py ×   ex2.1.py ×   ex_coelhos.py ×
1   ## dada uma sequecnia de fibonacci
2   ## 1 - 1 - 2 - 3 - 5 - 8
3   ## 0 - 1 - 2 - 3 - 4 - 5
4
5
6   def coelhos(n):
7       if(n == 1):
8           return 1
9       elif(n == 2):
10          return 1
11      else:
12          return coelhos(n-1) + coelhos(n-2)
13
14  k = int(input("Numero do 10 mes:"))
15  coelhos(k)
16  print(coelhos(k))
```

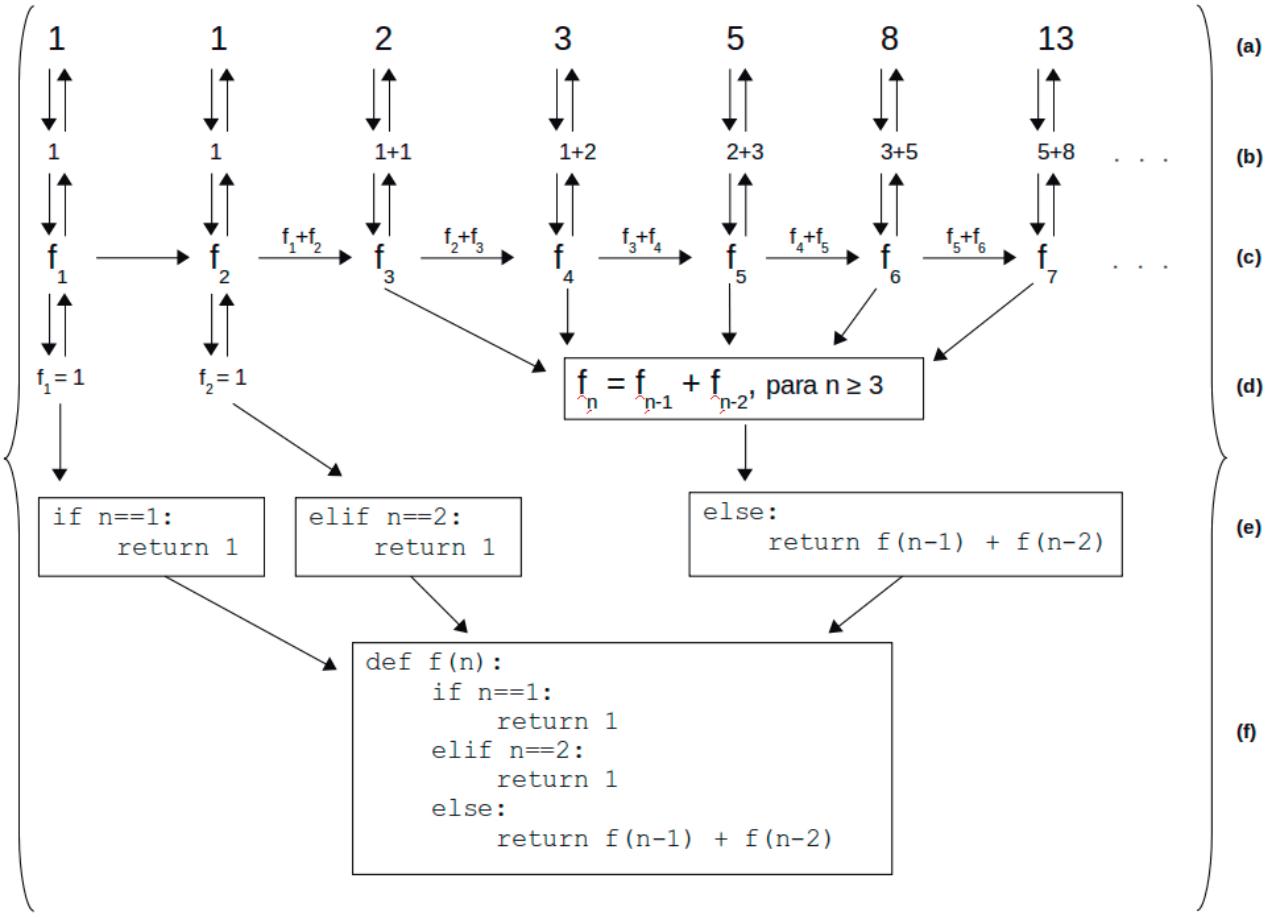
Fonte: Dados da Pesquisa.

Na solução desenvolvida pelo Grupo B (Figura 7), foi solicitado auxílio da professora para mostrar o resultado na tela (linha 16), pois estavam apenas chamando a função *coelhos(k)* e não estavam percebendo como deveriam fazer para mostrar o valor de retorno da função. A professora explicou a eles que deveriam usar o comando para mostrar o retorno da função na tela, tendo como argumento a chamada à função: *print(coelhos(k))*. Conseguiram realizar a alteração, fazendo com que o programa, com a função recursiva, mostrasse a resposta, como desejado.

VIDYA

VIDYA, v. 44, n. 2, p. 153-173, jul./dez. - Santa Maria, 2024.

Figura 8 - Rede de nodos estabelecida no desenvolvimento da atividade.



Fonte: Elaborado pela autora.

A rede de nodos com as diferentes representações, conceitos e suas relações construídas na atividade sobre Fibonacci está na Figura 8, de forma semelhante à Estrutura Conceitual Central proposta por Case (1992). Na linha (a) está a sequência numérica identificada pelos estudantes, ao resolverem o problema da reprodução dos coelhos, conforme apresentado na Atividade 4. Na linha (b), está a equivalência do elemento da sequência com a solução encontrada pelos estudantes, em que formaram os elementos como a soma de seus dois antecessores, para alcançar a generalização expressa na linha (d).

Na linha (c), a representação algébrica de cada elemento da sequência nomeado como f_n está associada, por meio das setas, à representação numérica da sequência e à generalização, na linha (d). A generalização é recursiva, pois na definição dos elementos a partir de $n = 3$, há a chamada da própria função. Na linha (e) estão representados, na linguagem de programação, cada um dos elementos da linha (d), a fim de compor o algoritmo recursivo. Na linha (f) está a função f em Python, implementada com a recursão, também a partir de n 3 e com os casos base para $f_1 = 1$ e $f_2 = 1$.

VIDYA, v. 44, n. 2, p. 153-173,
jul./dez. - Santa Maria, 2024.

As setas horizontais, na Figura 8, indicam o sentido do crescimento da sequência, em que cada coluna representa um mês e o número, na sequência, a quantidade de casais de coelhos no respectivo mês. A rede entre a representação numérica na linha (a), a decomposição dos números que está na linha (b) e a representação simbólica da linha (c) é ilustrada com as setas verticais com duplo sentido. A generalização expressa na linha (d) possui um único sentido de ligação que é na direção do algoritmo recursivo desenvolvido.

Essa atividade instigou os estudantes, por meio da investigação e desenvolvimento da definição recursiva e a partir dela, a construção do algoritmo recursivo. Como percebe-se nas relações da rede de nodos da Figura 8, a transformação da representação simbólica para a linguagem de programação Python é direta, acrescentando a estrutura de seleção da linguagem como para o caso de contemplar a geração dos elementos da sequência 1 e 2 e, posteriormente, a partir do terceiro elemento ($n \geq 3$).

As habilidades do Pensamento Computacional trabalhadas foram: abstração, algoritmos e procedimentos, coleção, análise e representação de dados e decomposição do problema, conforme a definição de CSTA, ISTE e NSF (2011). Correspondem às habilidades evidenciadas na atividade anterior, contudo o problema proposto constituiu-se por maior complexidade, exigindo que os estudantes buscassem os conceitos aprendidos e, a partir deles, elaborassem a solução.

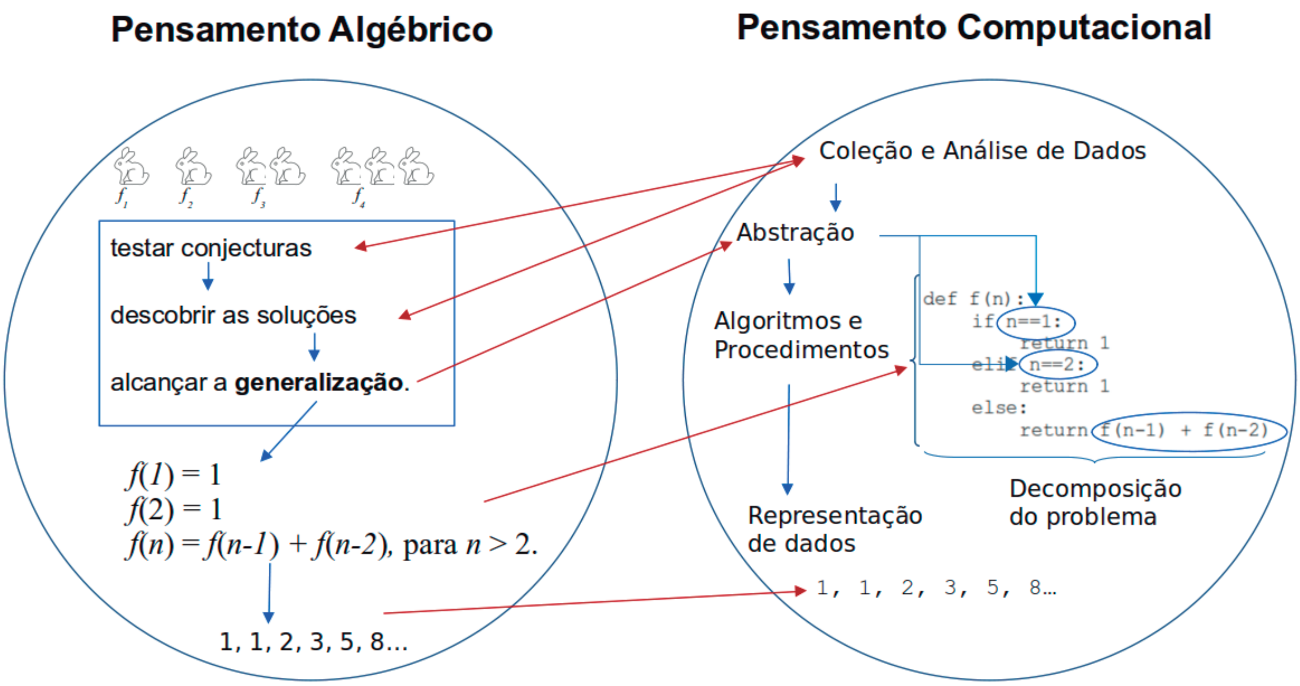
Nesse contexto, para testar as conjecturas e descobrir as soluções, os estudantes precisaram coletar os dados relevantes do problema, os quais foram apresentados de forma descritiva. Buscaram relações entre os dados, precisaram abstrair, testar conjecturas, até ser descoberto o padrão existente (análise de dados).

Para obterem a generalização, a abstração foi essencial, ao determinarem as equações da sequência de Fibonacci. A definição encontrada foi recursiva. A habilidade algoritmos e procedimentos, ao desenvolveram o algoritmo e implementá-lo para produzir a sequência numérica esteve presente. Como a função foi implementada de forma recursiva, o problema foi decomposto em partes menores, por exemplo, os dois casos base, quando $n = 1$ e $n = 2$ (decomposição de problema) e os demais elementos da sequência, sendo gerados recursivamente pela definição da soma de seus dois elementos antecessores.

A habilidade de Decomposição do Problema, possibilitou que os estudantes transferissem a representação simbólica desenvolvida por meio da generalização (Pensamento Matemático) e para a linguagem Python, na função recursiva (Pensamento Computacional). A representação da articulação do Pensamento Algébrico e do Pensamento Computacional está na Figura 9. A definição recursiva, no círculo do Pensamento Matemático, relaciona-se diretamente à implementação recursiva desenvolvida e presente no círculo Pensamento Computacional, da Figura 9 e que produziu a representação numérica da sequência de Fibonacci.

VIDYA, v. 44, n. 2, p. 153-173, jul./dez. - Santa Maria, 2024.

Figura 9 - Relações entre o Pensamento Algébrico e Pensamento Computacional.



Fonte: Elaborado pela autora.

O desenvolvimento das atividades propostas aos estudantes, propiciou um encadeamento dos conceitos em formação e que eles fossem gradativamente agregando condições de resolução aos problemas. A atividade sobre os números ímpares incentivou a recursão na definição matemática e no desenvolvimento do algoritmo e a outra atividade, com o desafio da recursão, porém para um problema laborioso, como a sequência de Fibonacci.

Na continuação, os estudantes foram instigados a criar problemas relacionados a padrões numéricos, com recursão. Para a criação dos problemas, foi solicitado que durante a realização, fossem identificados os elementos essenciais a todos problemas, propostos por Jurado (2018): (1) Informação; (2) Requisitos; (3) Contexto; (4) Conteúdo Matemático.

A informação sobre o problema trata dos dados e sua forma de apresentação, como por exemplo quantitativa, figurada ou relacional. Os requisitos referem-se ao que o problema está solicitando para que se encontre ou conclua. O contexto pode ser intra matemático, quando a abordagem de situações é no interior da matemática ou extra matemático quando extrapola este contexto, trazendo situações do cotidiano, por exemplo. Por fim, o conteúdo matemático abordado na resolução do problema (JURADO, 2018). Junto a estes aspectos, foi acrescentada a "Etapa do ensino", para que os grupos identificassem a etapa do ensino considerada adequada para aplicar o problema desenvolvido. Neste artigo, é discutido o problema que envolveu a recursão.

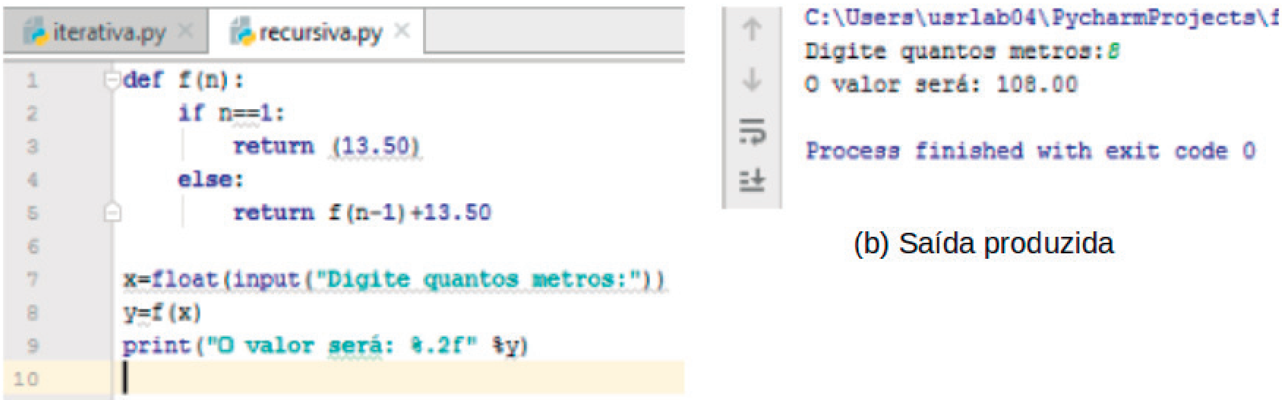
O problema, envolvendo a recursão, desenvolvido pelo grupo foi apresentado como: *"Joana deseja comprar 8 metros de um tecido com estampas de natal, para a mesa da ceia na noite natalina. Sabendo que cada metro do tecido custa R$ 13,50, quanto ela pagará pelos 8 metros de tecido?"* Juntamente com os elementos de Jurado (2018): Informação - dados relacionais; Requisitos -

pede-se o valor total de uma compra, dado o valor unitário da peça; Contexto: intra-matemático; Conteúdo: função afim; Etapa do ensino: ideal para o primeiro ano do ensino médio.

Faz-se uma observação no contexto, pois é extra matemático, traz um problema do dia-a-dia para solucionar. O grupo desenvolveu a solução computacional a este problema, por meio de um algoritmo recursivo implementado em Python (Figura 10). Esse aspecto foi considerado bastante relevante, na observação da pesquisa, o uso da recursão em diferentes problemas, como esse exemplo, em que há uma solução iterativa, que na maioria das vezes é encontrada trivialmente.

Na solução recursiva, o caso base $n = 1$, tem-se o valor de um metro de tecido 13,50. Para quantidades maiores de tecido, considerando que é vendido de metro em metro, tem-se a recursão em que a função *f* é invocada novamente, como na linha 5 da Figura 10: $f(n - 1) + 13{,}50$. A saída produzida na execução do programa desenvolvido está na Figura 10(b). Para a entrada 8, ou seja, a quantidade de metros de tecido, a saída produzida é 108,00, uma vez que um metro de tecido, conforme o enunciado do problema é 13,50 e cujo resultado foi calculado de forma recursiva.

Figura 10 - Solução recursiva ao Problema criado e saída produzida.



(a) Solução Iterativa

(b) Saída produzida

Fonte: Dados da Pesquisa.

A respeito dos processos reguladores de Case (1989), observou-se que o Grupo procurou verificar o que havia sido construído nas aulas, caracterizando o processo de Imitação. O processo Regulação Mútua aconteceu na interação entre os estudantes da dupla, para a criação e solução do problema e o processo regulador Resolução de Problemas, quando cada um procurou traçar as suas estratégias de resolução.

Além da atividade proposta, por meio deste problema, há outras potencialidades observadas que podem ser exploradas. Por exemplo, a formalização da definição recursiva para a solução, como: $f(n) = 13.50$, *se* $n = 1$ e $f(n) = f(n\text{-}1) + 13.50$, *se* $n > 1$. Outro aspecto que poderá ser associado ao problema, é a definição da função matemática para calcular o valor a pagar. Importante salientar fazer a observação que a implementação foi diretamente por meio do algoritmo recursivo na linguagem Python, demonstrando que “pensaram recursivamente” e obtiveram o resultado desejado.

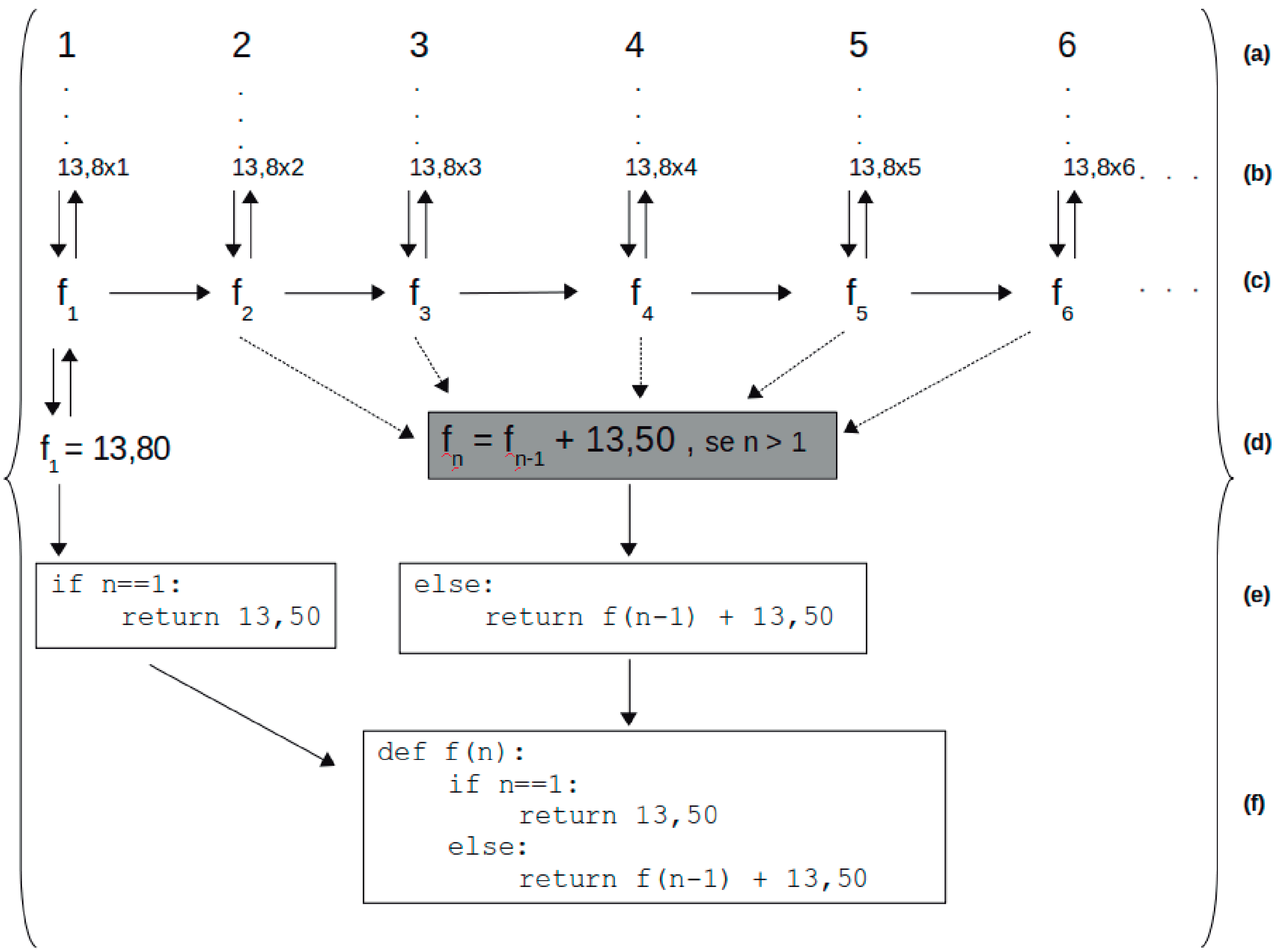
Os conceitos, representações e significados envolvidos no Problema proposto estão na rede de nodos elaborada (Figura 11). A partir do enunciado do problema, a quantidade de metros de tecido,

VIDYA, v. 44, n. 2, p. 153-173, jul./dez. - Santa Maria, 2024.

sendo esta pertencente aos números naturais, é representada na linha (a), como uma sequência numérica. Na linha (b), está o valor a ser pago, conforme a quantidade comprada, descrita no problema: para um metro o valor é 13,50 reais. A representação algébrica de cada elemento da sequência está na linha (c).

A representação da equação da generalização (de forma recursiva) da sequência está em cinza na linha (d), da Figura 11, porque não foi explicitamente definida pelo Grupo e sim, houve o raciocínio algébrico para chegar ao algoritmo recursivo (linhas (e) e (f)). Na linha (d), está a parte do algoritmo que corresponde à f_1, ligado ao primeiro termo da sequência e aos demais termos f_n, quando $n > 1$. Diretamente ligada a esta definição, na linha (e) estão as partes da função recursiva que correspondem aos termos da linha (d). Por fim, na linha (f) está a função recursiva implementada pelo Grupo A, na linguagem Python. As setas verticais, na Figura 11, representam o sentido da transição realizada entre os conceitos e as chaves externas representam a integração de todos os elementos e suas relações.

Figura 11 - Rede de nodos envolvendo o Problema criado.



Fonte: Elaborado pelos autores.

Houve evidências de terem sido trabalhadas as habilidades do Pensamento Computacional, consideradas aquelas conforme a definição das organizações *International Society for Tecnology in Education* (ISTE), *Computer Science Teachers Association* (CSTA) e *National Science Foundation* (NSF) (CSTA; ISTE; NSF, 2011). As habilidades foram Coleção, Análise e Representação de Dados (com a representação numérica e simbólica), Algoritmos e Procedimentos, Abstração e Decomposição do Problema (em problemas menores, como na recursão, até ser alcançada a solução geral do problema).

Os algoritmos iterativos aparentemente são intuitivos, mas a recursão foi incluída na resolução dos problemas, fazendo uma conexão com a definição de algoritmos recursivos e a recursão na matemática. A via realizada foi no sentido da resolução matemática, pela característica dos sujeitos serem licenciandos em Matemática.

Abstrair as características do padrão a fim de desenvolver o algoritmo que o gerou, com as diferentes formas de representação, foi uma constante no desenvolvimento das atividades.

Segundo Allevato e Onuchic (2019), uma das perspectivas a considerar, é o estabelecimento de conexões entre conteúdos de domínios distintos, ou seja, as relações feitas com os conteúdos da própria Matemática. Esse olhar a partir das conexões não está no escopo da investigação, porém percebeu-se que aconteceu ao longo do estudo de caso. Os estudantes estiveram envolvidos na resolução proposta e foi viabilizado a apreensão de significados aos objetos matemáticos, pelo estabelecimento de conexões, como é previsto na Base Nacional Comum Curricular (BNCC) (BRASIL, 2018), dando sentido aos conceitos trabalhados.

Nas atividades propostas, com sequência numérica, os estudantes trabalharam com o conceito de Progressão Aritmética e Função afim, como foi evidenciado nos artefatos produzidos e no relato dos estudantes, nas entrevistas. Neste contexto, observou-se o estabelecimento de conexões com outros conteúdos matemáticos.

Salienta-se a relação estabelecida entre o Pensamento Algébrico e o Pensamento Computacional pelos estudantes, o que se considera apropriada a abordagem no ensino, integrando esses pensamentos. Dessa forma, é possível tornar concreto o que foi generalizado, assim como perceber que com a generalização e a abstração, os algoritmos obtidos contemplam todos os elementos do padrão.

CONSIDERAÇÕES FINAIS

Ao apresentar o desenvolvimento do estudo de caso e refletir sobre o Pensamento Computacional, incluído no ensino dos licenciandos em Matemática, de forma transversal, foram construídos subsídios para inclusão da Computação na Educação, em vistas da qualificação de recursos humanos que futuramente atuarão na Educação Básica.

Os estudantes puderam resolver problemas e propor seus próprios problemas, no contexto do conteúdo matemático abordado, com a recursão. Evidenciou-se que os conceitos envolvidos, expressos na forma de Estrutura Conceitual Central, suas representações e relações estabelecidas compuseram uma rede de nodos interconectando a representação numérica, algébrica e computacional.

No estudo de caso desenvolvido, as habilidades do Pensamento Computacional foram associadas aos padrões e regularidades, conteúdo de álgebra, na Matemática, por meio da recursão.

Destaca-se o quão importante é a busca por regularidades e a obtenção da generalização. Isso envolve a abstração de características semelhantes que proporcionam a identificação da regularidade (Pensamento Algébrico), assim como a abstração, com pensamento recursivo, para o desenvolvimento do algoritmo, no que tange às formas de armazenamento, entrada e saída de dados e a estratégia da resolução empregada (Pensamento Computacional).

Os processos reguladores gerais de Case sustentaram o planejamento e favoreceram a prática pedagógica no decorrer da disciplina. Observou-se que o processo regulador Imitação acontecia especialmente ao ser trabalhado um conceito novo. De acordo com a progressão da construção do conhecimento, os outros processos reguladores surgiam. Os estudantes ao se depararem com um problema e buscarem alcançar seu objetivo (resultado desejado), precisavam estabelecer estratégias para alcançá-lo, caracterizando o processo Resolução de Problemas de Case.

O processo Exploração sempre esteve presente nas situações em que os estudantes, a partir de alguma estratégia conhecida, necessitavam descobrir novos caminhos, ainda não conhecidos, provocando a criação de novos objetivos. A interação entre os estudantes no desenvolvimento das atividades aconteceu, assim como a interação dos estudantes com a professora durante as aulas, caracterizando o processo Regulação Mútua e que proporcionou a troca de experiências, de ideias e a elaboração conjunta de estratégias para solução dos problemas.

Salienta-se que os estudantes reconheceram e aplicaram o Pensamento Computacional como uma forma de resolução de problemas, que pode ser associada ao ensino da Matemática. Para o desenvolvimento das soluções, além do conteúdo matemático, os estudantes precisaram adicionar o componente computacional, fazendo com que as formas de solução, em vários momentos, fossem distintas umas das outras e, mesmo assim, alcançado o objetivo. Isso instigou a uma flexibilidade na aceitação de diferentes soluções ao mesmo problema.

A generalização e alcançar a recursão não são processos simples e imediatos. Porém, foi evidenciado no estudo de caso, que são processos possíveis e viáveis no ensino. A partir da elaboração da definição matemática recursiva, para representar a generalização, foi notório que gradativamente, no decorrer das atividades, os licenciandos associaram-na à definição da recursão no algoritmo.

Considera-se que a linguagem Python favoreceu o trabalho, pois a representação simbólica matemática é muito próxima ou direta em relação à representação simbólica nesta linguagem, estando junto à realidade dos licenciandos. Por esse motivo, outro aspecto a investigar, é a recursão no sentido de partir da construção de algoritmos recursivos, alcançar a definição matemática recursiva.

Explorar o Pensamento Computacional com a recursão possibilitou aos futuros professores compreender e utilizar outra forma de resolução de problemas, além de promover o estabelecimento de relações entre conceitos da Matemática e da Computação.

A investigação realizada associou o ensino de Matemática e o Pensamento Computacional, embasados pela teoria de aprendizagem de Case. Nessa teoria, o desenvolvimento acontece de forma construtiva, promovido pela resolução dos problemas. Conforme o indivíduo vai resolvendo problemas cada vez mais complexos, novas estruturas são formadas. Tudo isso ocorre, amparado pelos conceitos que o indivíduo possui, os novos que vai construindo e o estabelecimento de relações semânticas entre eles e suas diferentes formas de representação, em determinado domínio do conhecimento, no caso, o domínio lógico-matemático.

Procurou-se alinhar o estudo de caso com a Base Nacional Comum Curricular (BNCC), que prevê o trabalho do Pensamento Algébrico com o Computacional, por meio da reflexão e abstração, associada à capacidade de transpor uma situação dada à diferentes maneiras de representação.

Acredita-se nessa forma transversal de integração do Pensamento Computacional à Matemática, como alternativa de trabalho, contextualizada com outra Ciência. A abordagem de resolução coerente ao contexto em que se vive favorece a preparação dos cidadãos, futuros professores,

VIDYA, v. 44, n. 2, p. 153-173,
jul./dez. - Santa Maria, 2024.

para atuar na sociedade em constante transformação. Nesta pesquisa, foi evidenciado que a integração entre conceitos matemáticos e computacionais permitiu uma nova maneira de trabalhar com o processo de formação inicial de professores de matemática.

REFERÊNCIAS

ACZEL, P. An introduction to inductive definitions. In: BARWISE, J. (Ed.). **Handbook of Mathematical Logic**. [S.l.]: Elsevier, 1977, (Studies in Logic and the Foundations of Mathematics, v. 90). p. 739 - 782.

ALLEVATO, N. S. G.; ONUCHIC, L. de la R. As conexões trabalhadas através da resolução de problemas na formação inicial de professores de matemática. **Revista de Ensino de Ciências e Matemática**, v. 10, n. 2, p. 01-14, 2019.

BLANTON, M. L.; KAPUT, J. J. Characterising a classroom practice that promotes algebraic reasoning. **Journal for Research in Mathematics Education**, v. 36, n. 5, p. 412-446, 2005.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**: educação é a base. Brasília, 2018. Disponível em: http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf. Acesso em: 10 mar. 2024.

BRASIL. Ministério da Educação. **Parecer CNE/CEB nº 2/2022**, de 17 de fevereiro de 2022. Brasil, 2022. Disponível em: https://www.computacional.com.br/docs_oficiais/parecer_homologado.pdf. Acesso em: 16 abr. 2024.

BRASIL. Ministério da Educação. **Resolução CEB 01/2022**, de 04 de outubro de 2022. Brasil, 2022. Disponível em: https://www.computacional.com.br/docs_oficiais/resolucao_ceb_012022.pdf. Acesso em: 16 abr. 2024.

CASE, R. **El desarrollo intelectual**: del nacimiento a la edad madura. Barcelona: Paidos, 1989.

CASE, R. **The Mind's Staircase:** Exploring the Conceptual Underpinnings of Children's Thought and Knowledge. Hillsdale: Lawrence Erlabaum Associates, 1992.

CORMEN, T. H. *et al.* **Introduction to Algorithms**. 2. ed. London: McGraw-Hill, 2001.

GODINO, J. D.; FONT, V. **Razonamiento algebraico y su didáctica para maestros**. Granada: Universidad de Granada, 2003.

CSTA; ISTE; NSF. **Computational Thinking Teacher Resource.** 2011. Disponível em: http://csta.acm.org/Curriculum/sub/CurrFiles/472.11CTTeacherResources_2ed-SP-vF.pdf. Acesso em: 20 abr. 2024.

GUNDLACH, B. H. **História dos números e numerais.** São Paulo: Atual, 1992.

JURADO, U. M. ¿cómo crear problemas de matemáticas?experiencias didácticas con profesores en formación. **Revista Iberoamericana de Educación Matemática (UNIÒN)**, n. 52, p. 307-313, abr. 2018.

NCTM. **Principles and Standards for School Mathematics.** Reston, VA: The National Council of Teachers of Mathematics, 2000

PEREIRA, M. C. N. **As investigações matemáticas no ensino-aprendizagem das sucessões:** uma experiência com alunos do 11o ano de escolaridade. 270 p. Tese (Doutorado)-Universidade da Beira Interior, Departamento de Matemática, Portugal, 2004. Coleção Teses. Associação de Professores de Matemática (APM).

ROMANO, F. **Learning Python**. Birmingham: PACKT Publishing, 2015.

RIBEIRO, L. *et al.* **Diretrizes de Ensino de Computação na Educação Básica.** Sociedade Brasileira de Computação, Relatório Técnico, n. 001, 2019.

VALE, I. As tarefas de padrões na aula de matemática: um desafio para professores e alunos. **Interacções**, n. 20, p. 181-207, 2012.

WING, J. M. Computational thinking. **Communications of the ACM**, ACM, New York, NY, USA, v. 49, n. 3, p. 33-35, mar. 2006.

WING, J. M. **Research Notebook**: Computational Thinking - What and Why? The Link. 2011. Pittsburgh, PA: Carneige Mellon. Disponível em: https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why. Acesso em: 19 abr. 2024.

YIN, R. K. **Estudo de caso**: planejamento e métodos. 5. ed. Porto Alegre: Bookman, 2015.