

Da Teoria à Prática: Resultados de uma Metodologia de Ensino de Pensamento Computacional para Pessoas Idosas

Bianca Maciel Medeiros¹, Francisco Carlos Nascimento de Lima¹, Rayana M. M. Cardoso¹, Sarah Carlas Gomes de Campos², Vinicius Eudes Galdino Fernandes¹, Isabel Dillmann Nunes¹

¹Instituto Metrópole Digital – Universidade Federal do Rio Grande do Norte (UFRN) – Natal, RN – Brasil

²Escola de Ciências e Tecnologia - Universidade Federal do Rio Grande do Norte (UFRN) – Natal, RN – Brasil

bel@imd.ufrn.br, bianca.medeiros.701@ufrn.edu.br,
rayanamendes00@hotmail.com, carlos.nascimento.710@ufrn.edu.br, vinicius.galdino.121@ufrn.edu.br, sarahcamposinfo@gmail.com

Abstract. *Computational Thinking is one of the skills necessary for solving problems in a logical, orderly and creative way. Elderly people must also be included in this world so that they can remain active and healthy, promoting not only digital autonomy in the consumption of information, but also as producers of digital technologies. Therefore, this article aims to present a methodology for teaching Computational Thinking, which begins with unplugged activities and ends with programming using digital tools, applied to a class of 14 elderly people. As a result, there is a gain in skills inherent to the learning process and mainly, the development of digital skills for today's world.*

Resumo. *O Pensamento Computacional é uma das competências necessárias para resolução de problemas de forma lógica, ordenada e criativa. As pessoas idosas também devem ser inseridas nesse mundo para que possam permanecer ativas e saudáveis, promovendo não somente uma autonomia digital no consumo de informações, mas também como produtoras de tecnologias digitais. Assim, este artigo tem o objetivo de apresentar uma metodologia de ensino de Pensamento Computacional, que inicia com atividades desplugadas e finaliza com a programação em ferramentas digitais, aplicado a uma turma de 14 idosos. Como resultado, tem-se o ganho de habilidades inerentes ao processo de aprendizagem e principalmente, do desenvolvimento de habilidades digitais para o mundo atual.*

1. Introdução

Atualmente, essa necessidade de estar inserido no mundo tecnológico vai muito além de apenas consumir tecnologia. A crescente demanda que é imposta aos indivíduos cotidianamente, torna cada vez mais exigido a capacidade de pensar, idealizar e produzir tecnologias, incluindo os indivíduos 60+. Assim, o conhecimento técnico deve ser aprendido para que as pessoas idosas possam também desenvolver soluções que sejam úteis para o seu dia a dia.

O declínio cognitivo pode ser considerado um dos problemas do envelhecimento, sendo reforçado pela dificuldade de se inserir tecnologicamente no mundo [Pinto 2010]. Assim, o ensino de Pensamento Computacional - PC, vem como uma alternativa de atividade para o estímulo cognitivo que, além de proporcionar a

atividade neural, contribui para que as pessoas idosas que exercem tais práticas se tornem cada vez mais funcionais na sociedade [Medeiros et al. 2024].

O ensino do Pensamento Computacional pode ser realizado por atividades desplugadas, que sem o uso das tecnologias digitais possibilitam o aprendizado dos seus pilares. O uso de ferramentas desplugadas, como o Desafio de Bebras¹, fazem com que os indivíduos que têm o contato vejam o Pensamento Computacional como algo mais palpável e acessível, principalmente no que diz respeito às pessoas idosas que enfrentam diversos desafios quando se trata do contato com tecnologias [Raymundo 2013].

Porém, o processo de inclusão digital se faz necessário e o uso de ferramentas desplugadas para introdução ao Pensamento Computacional pode ser uma ferramenta facilitadora, para que se possa alcançar as atividades plugadas [Brackmann 2017]. Oliveira et al. (2023) propõem uma metodologia para o ensino de Pensamento Computacional utilizando apenas atividades plugadas. Apesar de ter-se obtido uma boa resposta em relação à análise e resolução de problemas, foi percebido também que os idosos apresentaram certa dificuldade quanto ao uso do computador e da ferramenta de programação Scratch².

A inserção de pessoas idosas em atividades plugadas de Pensamento Computacional deve, então, iniciar com o aprendizado do uso de computador e suas ferramentas. Nesse sentido o ProEIDI (Projeto de Extensão de Inclusão Digital para Idosos) [Macêdo *et al.* 2023] desenvolveu um curso de Pensamento Computacional visando reduzir os receios associados ao uso de tecnologias.

Assim, este trabalho tem o objetivo de apresentar um curso de Pensamento Computacional realizado com pessoas idosas utilizando ferramentas desplugadas e plugadas e os seus resultados obtidos. Ao final da aplicação do método foi observado que a introdução ao PC despertou uma forma diferente de análise por parte dos indivíduos, fazendo com que no decorrer do curso eles estivessem utilizando os pilares do Pensamento Computacional para gerenciar até mesmo suas atividades cotidianas, evidenciando um bom estímulo cognitivo. Além disso, foi observado que a linguagem de programação textual possui uma melhor adesão quando se trata de programação para pessoas idosas, pois as situações apresentadas utilizando o ambiente de programação conseguem ser visualizadas de forma mais clara e aproximada das atividades desplugadas.

Este artigo está organizado da seguinte forma: Referencial teórico utilizado para a temática abordada no trabalho; Metodologia apresentada de acordo com a cronologia das atividades desenvolvidas; Exposição dos resultados obtidos e discussão sobre os mesmos; Apresentação das considerações finais acerca da aplicação do método.

2. Referencial Teórico

Este trabalho considera como conceitos essenciais o Envelhecimento Ativo e o Pensamento Computacional. A relação entre eles direciona os trabalhos realizados com

¹ <https://desafio.bebasbrasil.com.br/>

² <https://scratch.mit.edu/>

PC que permitem o aprendizado ativo da pessoa idosa sobre programação de software, não impossibilitando que sejam excluídas do mundo digital.

2.1 Envelhecimento ativo

Segundo a Organização Mundial de Saúde [World Health Organization 2015], a redução da taxa de fertilidade e o aumento da longevidade são fatores que contribuirão para o envelhecimento da população mundial. Esse envelhecimento populacional traz consigo desafios para os sistemas de saúde e previdência social. É importante ressaltar que envelhecer não implica necessariamente em adoecer, desde que não haja doenças associadas. Além disso, avanços na área da saúde e tecnologia têm proporcionado uma melhor qualidade de vida para a população que tem acesso a serviços adequados, sejam eles públicos ou privados. Portanto, é essencial investir em ações de prevenção ao longo de toda a vida [Miranda, Mendes & Silva 2016].

Diversos estudos abordam os problemas relacionados ao envelhecimento, tais como saúde [World Health Organization 2015], processo de ensino/aprendizagem [Wing 2006; Shute, Sun and Asbell-Clarke, 2017] e socialização e bem-estar [Lee and Park 2020]. O processo de envelhecimento deve ser considerado ao longo da vida e, por isso, a prevenção se torna fundamental. Ao agir dessa forma, é possível viver mais e com melhor qualidade de vida.

2.2 Pensamento Computacional

O Pensamento Computacional - PC é definido como um conjunto de habilidades e atitudes que podem ser aplicadas universalmente, despertando o interesse de não apenas cientistas da computação, mas de qualquer pessoa que tenha acesso ao seu ensino [Wing 2006]. Assim, o PC é dividido em 4 pilares: Decomposição (dividir o problema em problemas menores para facilitar a compreensão e solução); Abstração (reconhecer o que é mais significativo e deixar de lado o que não é essencial); Reconhecimento de Padrões (identificar problemas e situações semelhantes para auxiliar na solução) e Algoritmo (construir um sequência de passos para resolver o problema)[Shute, Sun and Asbell-Clarke 2017].

Desse modo, o ensino do Pensamento Computacional não significa adotar o raciocínio de uma máquina, mas envolver-se em processos mentais focados na solução eficaz e inovadora de problemas, incluindo o uso de abordagens algorítmicas para resolução óbices, a capacidade de navegar por distintas categorias da abstração e representação, trazendo a proximidade das estruturas de decomposição, fazendo a desassociação de interesses - o importante do desnecessário. Shute, Sun & Asbell-Clarke (2017) demonstram que o ensino do Pensamento Computacional reverbera na sociedade de diversas formas, podendo ser percebido por meio do foco na prestação de ações e serviços, a produção tecnológica própria, saindo do papel de consumidor e assumindo a função de produtor, sendo estes comportamentos atribuições do papel central à aprendizagem, permeados como meio de estabelecer relacionamentos e identificar fatores relevantes que possam gerar mudanças e impactos positivos.

Por isso, o ensino do Pensamento Computacional à pessoa idosa, além de incluí-la na sociedade tecnológica, a torna uma produtora de conhecimento e protagonista da sua própria evolução.

3. Metodologia

O curso de Pensamento Computacional faz parte do Projeto de Extensão de Inclusão Digital para Idosos (ProEIDI). O ProEIDI tem o objetivo de tornar o acesso da educação digital para a pessoa idosa no Rio Grande do Norte possível. Criado a partir da percepção da necessidade do ensino/aprendizagem sobre o uso de ferramentas digitais para o público 60+ por duas discentes do Bacharel em Tecnologia da Informação da Universidade Federal do Rio Grande do Norte, o projeto oferta aos participantes, atualmente, 4 cursos: Smartphone Básico, Smartphone Avançado, Computador e Pensamento Computacional. Os cursos ajudam a desenvolver habilidades essenciais para navegar no universo digital, desde o uso básico de dispositivos eletrônicos até a compreensão de conceitos mais complexos e criação de códigos em programas de software.

Dentre as modalidades de curso oferecidas pelo ProEIDI, destaca-se o curso de Pensamento Computacional, curso mais recente do projeto. O curso tem a duração de 8 semanas, com 1 encontro de 1h30min por semana, em laboratório de informática equipado com computadores compatíveis com o tipo de tecnologia usada. O laboratório utilizado conta com 32 computadores e equipamentos auxiliares, como projetor (tipo DataShow), caixa de som e quadro. O curso aconteceu no período de 13 de abril a 08 de junho deste ano. A metodologia também é direcionada para um aprendizado personalizado, em que a cada 2 idosos são acompanhados por um monitor, além de 2 professores responsáveis pelo conteúdo e atividades. Além disso, um material didático impresso, que consiste nos conteúdos da aula e exercícios, é entregue sempre que o assunto em questão é abordado em sala de aula.

Neste curso, os alunos aprendem conceitos básicos sobre o Pensamento Computacional, tais como decompor problemas em partes menores, identificar padrões, abstrair o que não é relevante e criar algoritmos. Além das aulas expositivas e práticas, o curso conta com material didático e exercícios adaptados para o público 60+, o qual estimula a prática mesmo fora do ambiente de sala de aula. Essas habilidades não só melhoram a compreensão tecnológica dos alunos, mas também fortalecem seu raciocínio lógico e capacidade de resolução de problemas em diversas áreas da vida.

O curso teve a participação de 14 idosos, sendo 9 com idade entre 60 e 65 anos, 3 idosos entre 66 e 70 anos e 2 idosos entre 71 e 80 anos. Do total de alunos, 9 eram do sexo feminino e 5 do sexo masculino, em sua maioria (quase 65%) com escolaridade de nível superior.

Os módulos do curso foram estruturados para atender tanto os conceitos teóricos quanto práticos, da seguinte forma:

- Teórico — conceitos de Pensamento Computacional. Essa parte possui as seguintes características:
 - 3 encontros;

- Revisar do uso de Computador;
- Apresentar e aprofundar os conceitos fundamentais do Pensamento Computacional;
- Atividades desplugadas - por exemplo, uso de uma pessoa "robô" que deve obedecer os comandos dos colegas, para aprendizagem do conceito de Algoritmo;
- Atividades plugadas - por exemplo, uso da ferramenta para responder um questionário sobre o tema.
- Prático — criação de códigos em programa de software. Essa parte possui as seguintes características:
 - 5 encontros;
 - Introdução aos conceitos de Lógica de Programação;
 - Atividades desplugadas - por exemplo, a escrita de passo a passo de quem poderia votar em uma eleição de forma colaborativa;
 - Atividades plugadas - por exemplo, escrita de algoritmo em uma linguagem de programação.

De forma detalhada, o Quadro 1 apresenta cada encontro considerando os tópicos abordados, o objetivo da aula, o aprendizado e o resultado.

Quadro 1 - Detalhamento das aulas (Pensamento Computacional)

Aula 1 - Revisão (Atividade prática) e introdução ao Pensamento Computacional	
Tópicos abordados	Revisão sobre o manuseio do computador e utilização do Google Drive e Gmail. Apresentação geral dos pilares do Pensamento Computacional (Decomposição, Abstração, Reconhecimento de padrões e algoritmos).
Objetivo	Verificar a lembrança e familiaridade dos indivíduos para com as tecnologias necessárias para desenvolver as atividades do curso; realizar uma introdução ao primeiro assunto do conteúdo programado (Pilares do Pensamento Computacional)
Aprendizado	Compreensão sobre o conceito geral dos pilares do pensamento computacional e acerca da presença do Pensamento Computacional no cotidiano
Resultado	Discussões interativas a respeito da presença do conceito de Pensamento Computacional e sua presença no dia-a-dia, com participação e exemplificações, mostrando a compreensão acerca da teoria apresentada a partir da capacidade de fazer analogias.
Aula 2 - Continuação da explicação sobre os pilares do Pensamento Computacional	
Tópicos abordados	Pilares do Pensamento Computacional (Decomposição e abstração).
Objetivo	Reforçar por meio de explicação oral os conceitos dos pilares do Pensamento Computacional (Abstração e decomposição) verificar a assimilação do conteúdo por meio de atividades propostas.
Atividades propostas	Cada aluno escolheu uma receita culinária ou atividade de sua preferência para descrever de forma numerada (decompondo a atividade em partes menores) e organizar em uma sequência de "passos" (Abstraindo as partes não necessárias em cada passo distinto).

Materiais utilizados	Caneta e papel (Atividade desplugada)
Aprendizado	Compreensão específica sobre Decomposição e Abstração e fixação a partir de exercícios práticos.
Aula 3 - Continuação da explicação sobre os pilares do Pensamento Computacional	
Tópicos abordados	Pilares do Pensamento Computacional (Reconhecimento de padrões e Algoritmos).
Objetivo	Reforçar por meio de explicação oral os conceitos dos pilares do Pensamento Computacional (Reconhecimento de padrões e Algoritmo) verificar a assimilação do conteúdo por meio de atividades propostas.
Atividades propostas	1- Atividade de fixação sobre reconhecimento de padrões (Atividade plugada com uso do questionário " <i>Padrões e sequências</i> " encontrado na plataforma "Wonderwall ³ ") 2- Atividade desplugada em grupo onde foi construído um algoritmo que mostrasse a sequência de passos necessária para retirar um "robô" desprovido de conhecimento do laboratório onde ocorreram as aulas. Nessa atividade, os idosos ditavam as instruções a serem seguidas pelo "robô" e era executado o teste, caso houvesse algum erro de execução decorrente da falta/erro na informação fornecida, o "robô" voltava para a posição inicial.
Materiais utilizados	Computador (Atividade plugada)
Aprendizado	Compreensão específica sobre Reconhecimento de padrões e Algoritmos e fixação a partir de exercícios práticos.
Aula 4 - Introdução à lógica computacional	
Tópicos abordados	O que é lógica computacional; para que utilizamos a lógica computacional; introdução aos conceitos do da lógica computacional.
Objetivo	Explicar o conceito de lógica computacional e sua usabilidade; explicar os conceitos de entrada de dados, variáveis, estruturas condicionais e estruturas de repetição
Atividades propostas	1- Atividade grupal (Escrita de um algoritmo utilizando os conceitos aprendidos sobre lógica computacional). Foi dada uma situação onde era necessário a criação de um sistema que identificasse se uma pessoa era obrigada a votar ou não. A partir disso surgiu uma discussão sobre como um algoritmo com esse intuito de funcionamento poderia ser estruturado.
Materiais utilizados	Quadro branco e piloto.
Aprendizado	Primeiro contato sobre estruturação de algoritmos sem utilizar uma linguagem de programação propriamente dita, partindo dos conhecimentos adquiridos sobre lógica somados à intuição de como a sequência de instruções poderia ser escrita.
Aula 5 - Revisão dos conteúdos abordados anteriormente e exercício	

³ <https://wordwall.net/>

Tópicos abordados	Revisão dos conteúdos abordados sobre pilares do Pensamento Computacional e lógica computacional; Primeiro contato com a linguagem de programação estruturada a partir de exercícios.
Objetivo	Sanar dúvidas sobre os conteúdos abordados até o momento (pilares do PC e Lógica computacional) e introduzir a linguagem de programação estruturadas por meio de jogo interativo (Atividade plugada).
Atividades propostas	Exercício individual utilizando plataforma <i>computer it</i> ⁴ .
Materiais utilizados	Computador.
Aprendizado	Primeiro contato com a linguagem de programação estruturada. Com a utilização do jogo onde é necessário fazer a interpretação e ter a compreensão da lógica estabelecida, foi possível que os indivíduos tivessem uma maior noção de como funcionam os algoritmos.
Aula 6 - Introdução a linguagem de programação	
Tópicos abordados	O que é linguagem de programação; Introdução ao ambiente de programação Égua ⁵ ; Como escrever um texto utilizando a linguagem de programação Égua; O que é uma variável; Como declarar uma variável utilizando a linguagem de programação Égua; Como escrever o conteúdo de uma variável.
Objetivo	Explicar o que é uma linguagem de programação; introduzir a linguagem de programação Égua; Explicação da sintaxe utilizada para exibir textos, criar variáveis e exibir conteúdos de variáveis.
Atividades propostas	A cada explicação de um tópico foi proposta uma atividade para ser desenvolvida dentro do ambiente de programação.
	1- A primeira atividade consistiu em construir um algoritmo que exibisse um texto escolhido de forma livre.
	2- Na segunda atividade foi solicitado que fosse criada uma variável e atribuir um conteúdo a ela.
	3- Na terceira atividade foi solicitado que fosse criado um algoritmo que exibisse o conteúdo da variável criada no exercício 2.
Materiais utilizados	Computador.
Aprendizado	Primeiro contato com a linguagem de programação égua, gerando diversas discussões sobre o conceito de linguagem de programação. Teoria aplicada de forma dinâmica e com uso de muitas analogias o que gerou bastante interação. Para prática e fixação dos conteúdos abordados foram realizadas com sucesso atividades práticas.
Aula 7 - Introdução a linguagem de programação (Continuação)	
Tópicos abordados	Revisão (Como exibir um texto utilizando a linguagem Égua, conceito de variáveis, como declarar variáveis utilizando a linguagem de programação Égua e como exibir o conceito de variáveis); Explicação teórica sobre

⁴ <https://compute-it.toxicode.fr/>

⁵ <https://egua.dev/>

	operadores matemáticos (Aritméticos e relacionais); Prática (Operadores matemáticos).
Objetivo	Relembrar os conceitos abordados na Aula 6; Explicar teoricamente o conceito de operadores e para que serve cada um deles; Explicar a sintaxe adequada para utilizar operadores matemáticos dentro da linguagem de programação; Realizar atividades práticas utilizando os operadores vistos na teoria.
Atividades propostas	A cada explicação de um tópico foi proposta uma atividade para ser desenvolvida dentro do ambiente de programação.
	1- A primeira atividade realizada após a explicação teórica sobre operadores matemáticos (aritméticos), solicitou que fosse criado um algoritmo que realizasse as seguintes operações: Soma; subtração; multiplicação; divisão e resto da divisão. 2- Na segunda atividade, realizada após a explicação teórica sobre operadores matemáticos (Relacionais), foi solicitado que fosse criado um algoritmo que verificasse se um número é maior que outro, menor que outro, igual a outro ou diferente de outro.
Materiais utilizados	Computador, lápis e papel.
Aprendizado	Primeiro contato com operadores matemáticos dentro da linguagem de programação. Foi visto e praticado como construir expressões matemáticas dentro do ambiente de programação e fazer verificações entre valores. Nas atividades propostas (plugadas) acabaram sendo utilizadas ferramentas desplugadas (lápis e papel) pois os indivíduos sentiam mais segurança em implementar o algoritmo construindo a lógica de forma escrita no papel antes de passar o conteúdo para o ambiente de desenvolvimento.
Aula 8 - Linguagem de programação (Continuação) e Estrutura condicional.	
Tópicos abordados	Revisão (Operadores matemáticos aritméticos e relacionais); Explicação teórica sobre operadores lógicos ("e" e "ou"); Explicação teórica sobre estruturas condicionais dentro do ambiente de programação; Prática (Projeto utilizando os conteúdos vistos durante o curso).
Objetivo	Relembrar os conceitos abordados na Aula 7; Explicar teoricamente o conceito dos operadores lógicos abordados ("e" e "ou") e para que serve cada um deles; Explicar a sintaxe adequada para utilizar operadores lógicos dentro da linguagem de programação; Explicar de forma teórica estrutura condicional utilizando a linguagem Égua; Testar os conhecimentos adquiridos durante o curso por meio de uma atividade prática.
Atividades propostas	Após a explicação teórica dos conteúdos abordados na aula 8, foram feitas três propostas de projetos (cada uma com um nível de dificuldade diferente) para pôr em prática a combinação dos conhecimentos adquiridos durante o curso. Os indivíduos deveriam escolher um dos projetos propostos e construir o algoritmo solicitado.
	Proposta 1- Crie um algoritmo que dirá se um número é ímpar ou par.
	Proposta 2- Crie um algoritmo que dirá se uma pessoa pode dirigir ou não e se ela é habilitada.
	Proposta 3- Crie um algoritmo que indica se uma pessoa tem idade para votar, se não tem idade para votar e se ela é obrigada a votar ou não.

Materiais utilizados	Computador.
Aprendizado	Primeiro contato com operadores lógicos dentro da linguagem de programação. Foi visto como construir expressões lógicas dentro do ambiente de programação. Também, foi o primeiro contato com uma estrutura lógica mais complexa (estrutura condicional) utilizando a linguagem de programação.

O intuito das aulas descritas é permitir que os idosos possuam um aprendizado colaborativo e de forma a evoluir na construção do conhecimento em relação aos pilares de Pensamento Computacional. Ao fim do curso, as aulas práticas e voltadas para a construção de algoritmos executáveis em uma linguagem de programação proporcionam a experiência necessária para o real entendimento de como um computador funciona.

4. Discussão dos resultados

No decorrer das aulas descritas no Quadro 1, foi observada a evolução dos idosos em relação aos conceitos de Pensamento Computacional, eles adquiriram novos conhecimentos e a capacidade de “pensar computacionalmente”, isso devido ao entendimento acerca dos conteúdos ministrados.

A mesclagem de atividades plugadas e desplugadas foi um fator contribuinte para o melhor aproveitamento dos idosos em relação às aulas. Foi observado que a assimilação dos conteúdos era mais efetiva quando eles conseguiam fazer analogias entre o que tinham feito nas atividades desconectadas e o que estava sendo pedido nas atividades conectadas.

A Figura 1 mostra a turma realizando atividades plugadas de desenvolvimento de raciocínio lógico. É importante ressaltar que as atividades permitiam tanto o desenvolvimento do conceito relacionado como também do aprimoramento das habilidades no computador e no desenvolvimento cognitivo das pessoas idosas.



Figura 1. Alunos do curso de PC realizando atividades de Raciocínio Lógico.

Ao final do curso foi realizada uma pesquisa para entender quais melhorias deveriam ser realizadas, como também verificar com o que as pessoas idosas participantes se identificaram mais. Porém, devido ao curto tempo do curso, somente 7 participantes responderam, o que permite identificar *insights* para a reflexão de cursos futuros.

Um dos pontos a chamar atenção, foi a percepção dos idosos sobre o seu aprendizado, comparando o seu conhecimento antes e depois do curso. A Figura 2 mostra o conhecimento antes do curso, enquanto a Figura 3 mostra o aprendizado após o curso.

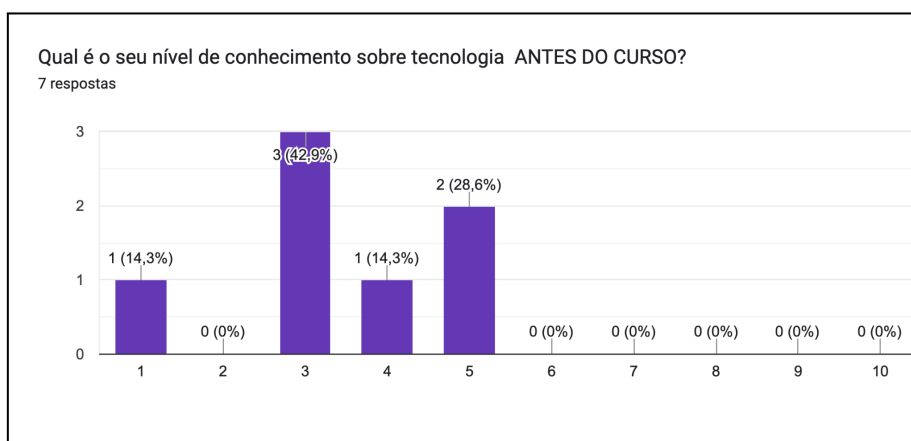


Figura 2. Nível de conhecimento antes do curso de PC.

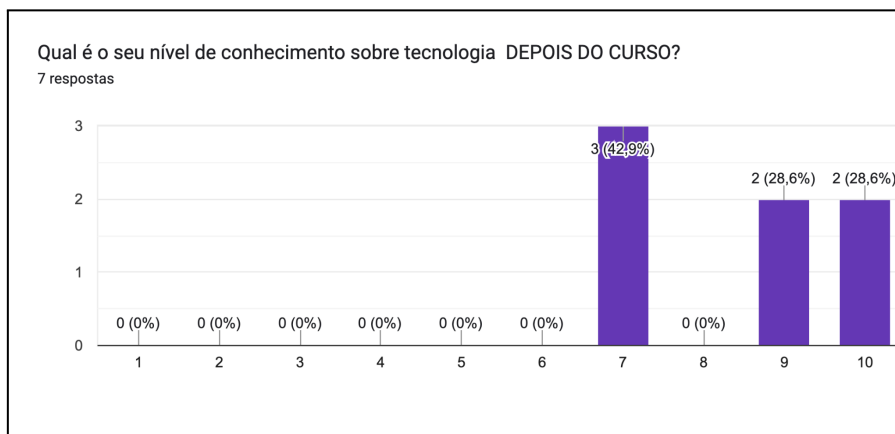


Figura 3. Nível de conhecimento depois do curso de PC.

A percepção de evolução no nível de conhecimento percebe-se também nos códigos construídos pelos idosos durante o curso. A Figura 4 mostra a evolução da aprendizagem de programação utilizando a ferramenta ÉGUA. É possível identificar a complexidade do código no último dia do curso, em que os alunos construíram um pequeno projeto como atividade final.

Primeiro contato com o Egua

Exemplos

```
1 escreva("Projeto de inclusão digital para idosos");
2 escreva ("Plano de Comunicação Digital");
```

Executar

Projeto de inclusão digital para idosos
Plano de Comunicação Digital

Exemplos

```
1 escreva("Projeto de inclusão digital para idosos");
2 escreva ("Plano de Comunicação Digital");
3 var idadeNicolau = 70;
4 escreva("qual a idade de Nicolau?");
5 escreva(idadeNicolau);
6 var anosproeidi = 3;
7 escreva("quantos anos de proeidi?");
8 escreva(anosproeidi);
```

Executar

Projeto de inclusão digital para idosos
Plano de Comunicação Digital
qual a idade de Nicolau?
70
quantos anos de proeidi?
3

Segunda aula de programação - Operadores lógicos

Exemplos

```
1 var num1 = 60;
2 var num2 = 300;
3 escreva("A SOMA É");
4 escreva(num1 + num2);
5 escreva("A SUBTRAÇÃO É");
6 escreva(num1 - num2);
7 escreva("A MULTIPLICAÇÃO É");
8 escreva(num1 * num2);
9 escreva("A DIVISÃO É");
10 escreva(num1 / num2);
11 escreva("MÓDULO É");
12 escreva(num1 % num2);
13 escreva("MAIOR QUE");
14 escreva(num1 > num2);
15 escreva("MENOR QUE");
16 escreva(num1 < num2);
17 escreva("IGUALDADE");
18 escreva(num1 == num2);
19 escreva("DIFERENTE");
20 escreva(num1 != num2);
```

Executar

A SOMA É
360
A SUBTRAÇÃO É
-240
A MULTIPLICAÇÃO É
18000
A DIVISÃO É
0.2
MÓDULO É
60
MAIOR QUE
falso
MENOR QUE
verdadeiro
IGUALDADE
falso
DIFERENTE
verdadeiro

Projeto final - algoritmo para saber se aluno é apto inscrição no Proeidi

Exemplos

```
1 escreva ("verificação da inscrição PROEIDI");
2 escreva ("-----");
3 var anonascimento = 2000;
4 var anoatual = 2024;
5 se(anoatual - anonascimento >= 60){
6 escreva("aluno apto à inscrição");
7 }
8 senão {
9 escreva ("aluno não apto");
10 }
11
```

Executar

verificação da inscrição PROEIDI

aluno não apto

Figura 4. Evolução nos códigos utilizando a ferramenta ÉGUA.

Além dessa percepção da própria evolução do aprendizado pelos idosos, outros pontos foram positivos: 100% dos respondentes consideraram o material didático ótimo ou muito bom; e 100% dos respondentes consideraram a atuação dos professores e monitores de forma ótima ou muito boa. Porém, alguns pontos foram levantados para serem melhorados, como a quantidade de aulas (57% consideraram insuficientes) e sua duração de 1h30min (37% consideraram insuficientes).

Tal perspectiva colabora para que o curso seja considerado bem formatado, com tecnologias adequadas para o público da pessoa idosa e com a atenção necessária dos professores e monitores. Porém, o tema é complexo e deve ser tratado de forma mais detalhada e com uma quantidade de exercícios maior.

5. Considerações finais

A proposta do curso de Pensamento Computacional apresentado neste artigo traz a evolução do aprendizado de pessoas idosas no uso e na construção de tecnologias digitais. A necessidade do entendimento de tais recursos é necessário para o seu cotidiano, porém, além disso, a criticidade em relação às tecnologias e o conhecimento de como são desenvolvidas se faz presente nas atividades desse público que permanece ativo, sociável e atuante no mundo do trabalho.

O curso está vinculado ao ProEIDI - Projeto de Extensão de Inclusão Digital do Idoso do Instituto Metr pole Digital da Universidade Federal do Rio Grande do Norte, permitindo que o idoso estabeleça uma trilha de aprendizado. Essa trilha inicia com o aprendizado de conhecimentos b sicos do uso do computador para que depois possa entender como os programas s o criados e assim, tamb m desenvolv -los.

A metodologia do curso foi definida para que as pessoas idosas tivessem a experi ncia real na cria o de programas, come ando com quest es te ricas at  aplic -las em problemas reais. O uso de atividades desplugadas e plugadas, tamb m permitiu uma intera o aprofundada entre a turma, de forma colaborativa, como tamb m com o computador. Por fim, o uso de um material did tico impresso, vi vel para acompanhamento durante as aulas e a realiza o de estudos e exerc cios fora da sala de aula, melhorou o andamento do curso, fornecendo uma evolu o no aprendizado.

Os estudos nesse sentido devem ser continuados para que melhorias possam ser aplicadas. O aumento da quantidade de exerc cios, o aumento de per odo do curso e assim o avan o nos conceitos de programa o devem ser testados e avaliados em uma turma pr xima. Al m disso, como trabalho futuro, criar o curso de Pensamento Computacional 2, em que conceitos de Intelig ncia Artificial tamb m s o estudados, exercitados e aplicados pelas pessoas idosas.

Refer ncias

- Brackmann, C. P. (2017) "Desenvolvimento do Pensamento Computacional atrav s de atividades desplugadas na Educa o B sica" 226 f. Tese (Doutorado em Inform tica na Educa o) - Centro Interdisciplinar de Novas Tecnologias na Educa o (CINTED), Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre.
- IBGE – Instituto Brasileiro de Geografia e Estat stica. (2018). "Proje o da Popula o 2018: n mero de habitantes do pa s deve parar de crescer em 2047". <https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/21837-projecao-da-populacao-2018-numero-de-habitantesdo-pais-deve-parar-de-crescer-em-2047>.
- Isbell, C. L., Stein, L. A., Cutler, R., Forbes, J., Fraser, L., Impagliazzo, J., Proulx, V., Russ, S., Thomas, R. e Xu, Y. (2009). "(Re)defining computing curricula by (re)defining computing". *ACM SIGCSE Bulletin*, 41(4), 195–207.
- Lee, E. and Park, S. (2020). "Immersive Experience Model of the Elderly Welfare Centers Supporting Successful Aging". *Frontiers in Psychology*, 11: 8.

- Macêdo, K. da P. F. de, Queiroz, P. M. G. de, Araújo, P. V. B., Cardoso, R. M. M., Campos, L. C. e Nunes, I. D. (2023). "Inclusão Digital para Idosos: evolução e percepções.", 43 Congresso da Sociedade Brasileira de Computação, 2023, João Pessoa.
- Medeiros B. M., Lima F. C. de N. e., Cardoso, R. M. M., Campos S. C. G. de., Fernandes V. E. G., Nunes, I. D., (2024) "Desenvolvimento cognitivo da pessoa idosa por meio de um curso de Pensamento Computacional", In: 1º Congresso Brasileiro sobre Envelhecimento Ativo e Saudável e XVI Seminário Internacional de Atividade Física para a Terceira Idade (CONBEAS).
- Miranda, G. M. D., Mendes, A. C. G. e Silva, A. L. A. (2016). "O envelhecimento populacional brasileiro: desafios e consequências sociais atuais e futuras", Revista Brasileira de Geriatria e Gerontologia, 19(3), 507-519.
- Oliveira Jr., E. R., Bortoli, L. A., Marchi, A. C. B., Pasqualotti, A. e GIL, H. T. (2023) "Proposição de uma sequência didática baseada no Pensamento Computacional para idosos", Simpósio Brasileiro de Informática na Educação (SBIE), Passo Fundo/RS. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, p. 1536-1545.
- Pinto, L. D. (2010) "Estudo de complexos de cobre com aminoácidos de interesse para a química do cérebro" Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio).
- Raymundo, T. M. (2013) "Aceitação de tecnologias por idosos", 89 f. Dissertação (Mestrado) – Escola de Engenharia de São Carlos/Faculdade de Medicina de Ribeirão Preto/Instituto de Química de São Carlos, Universidade de São Paulo, São Paulo, 2013.
- Wing, J. M. (2006). "Computational Thinking", Communications of the ACM, 49(3), 33-35.
- World Health Organization. (2015). "World report on ageing and health" World Health Organization. <https://apps.who.int/iris/handle/10665/186463>.