

# Estudantes Ensinando Computação para a Comunidade: Uma Experiência através do TISP

Camille L. Jesus  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
camillejesus27@gmail.com

Diego C. Silva  
PET Engenharías – MEC  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
diego.cs282@gmail.com

Francisco T. S. S. Pereira  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
francisco.p@gmail.com

Eduarda C. Oliveira  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
eduarda59costa60@gmail.com

Daniel A. Costa  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
dancostafsa@gmail.com

Laís L. C. Baptista  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
lais.lara.baptista@gmail.com

Ozenilson A. P. Cruz  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
ozenilsonalisson@gmail.com

Roberto A. Bittencourt  
UEFS – Universidade Estadual de  
Feira de Santana  
Feira de Santana, Bahia, Brasil  
roberto@uefs.br

## RESUMO

As tecnologias do século XXI e suas relações com a sociedade transformam o meio sociocultural e, por consequência, interferem no âmbito educacional. Deste modo, faz-se necessário o estímulo às iniciativas que discutam o uso de tecnologias atuais no meio escolar. Assim, este artigo deseja, através de um relato de experiência, encorajar grupos universitários no contexto de computação a desenvolverem práticas que aproximem a comunidade na qual estão inseridos às temáticas que abordem o uso e desenvolvimento de habilidades computacionais, apresentando as vivências e percepções de estudantes que integram o Programa de Professor em Serviço (TISP) na Universidade Estadual de Feira de Santana (UEFS). O artigo contempla o desenvolvimento e prática de oficinas de Arduino, Python e Informática básica para comunidades do interior da Bahia em nível universitário e pré-universitário. Percebeu-se no decorrer do estudo que, mediante a atuação docente, pôde-se desenvolver autonomia e criticidade nos participantes do grupo TISP, uma vez que estes planejavam e ministravam as aulas adequando-as às condições necessárias e contornando as situações adversas.

## CCS CONCEPTS

• **Social and professional topics** → **Informal education**; *CS1*.

---

Fica permitido ao(s) autor(es) ou a terceiros a reprodução ou distribuição, em parte ou no todo, do material extraído dessa obra, de forma verbatim, adaptada ou remixada, bem como a criação ou produção a partir do conteúdo dessa obra, para fins não comerciais, desde que sejam atribuídos os devidos créditos à criação original, sob os termos da licença CC BY-NC 4.0.

*EduComp'21, Abril 27–30, 2021, Jataí, Goiás, Brasil (On-line)*

© 2021 Copyright mantido pelo(s) autor(es). Direitos de publicação licenciados à Sociedade Brasileira de Computação (SBC).

## PALAVRAS-CHAVE

educação em engenharia, inclusão digital, TISP, aprendizagem de programação

## 1 INTRODUÇÃO

Vivemos na era digital. Nesta era, o campo da educação vêm sofrendo constantes alterações a fim de adaptar o ensino para o contexto sociocultural dos estudantes do século XXI. Nesta nova realidade, é preciso incluir as tecnologias de informação e comunicação pelo papel relevante que assumem no ensino e no desenvolvimento dos sujeitos. Esta educação contextualizada em um mundo tecnológico e contemporâneo necessita de práticas que desenvolvam nos estudantes as capacidades fundamentais para o século XXI: pensamento crítico, criatividade, resolução de problemas, colaboração e comunicação [51]. Tais capacidades podem ser geradas mediante a exploração das tecnologias e permitir que os sujeitos adquiram a competência de “viver em um mundo multifacetado, com cidadãos ativos e responsáveis” [48]. Por isso, o uso de tecnologias em sala de aula vai além da limitação das práticas do uso de “ferramentas educacionais”, pois as tecnologias utilizadas no ensino “modificam os ambientes culturais e educativos, criam novos modos de comunicação e reformulam os papéis que as pessoas desempenham habitualmente” [21], além de proporcionar o desenvolvimento dos sujeitos mediante a interação com as tecnologias atuais, que, de outra forma, poderiam tornar-se “indivíduos obsoletos” [11].

Assim, ao considerarmos a tecnologia para a formação e desenvolvimento do sujeito, a computação passou a ter um papel relevante neste processo, uma vez que esta se faz presente nos processos tecnológicos envolvidos no cotidiano da humanidade [12]. Resnick afirma que a ciência da computação dispõe de possibilidades que utilizam de artefatos tecnológicos presentes na sociedade para o processo de ensino-aprendizagem, causando um impacto

positivo na educação, incluindo a motivação dos participantes envolvidos no processo e atuando como um recurso educativo capaz de desenvolver o conhecimento [43].

Entretanto, o ensino da computação para a comunidade brasileira externa aos cursos de graduação desta área ainda se encontra em estágio incipiente, com experiências pontuais, nem sempre com continuidade [25, 40]. Isto gera um déficit de ensino desta ciência na educação básica da nossa sociedade, muitas vezes se prolongando para o nível acadêmico. Neste contexto, diversos trabalhos [7, 13, 18, 19, 29] promovem, através do ato de desenvolvimento, prática e divulgação, o interesse da comunidade para com temáticas da computação. Tais práticas evidenciam que é possível encorajar e motivar estudantes a se aproximarem do campo da computação.

Para que isso se realize mais facilmente, o Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) promove, através de ramos estudantis, diversas atividades que estimulam o desenvolvimento científico de forma ampla e situada por meio de subgrupos denominados por “Ligas”. A Universidade Estadual de Feira de Santana (UEFS) possui um ramo ativo desde o ano de 2017, o IEEE UEFS. Este ramo busca, mediante suas atividades, o aperfeiçoamento dos membros e a divulgação dos conhecimentos e tecnologias desenvolvidos em suas ligas para a comunidade externa. Uma destas ligas, o TISP UEFS, surgiu em 2019, após o entusiasmo gerado por uma experiência de ensino realizada no ano anterior, por membros do ramo estudantil na cidade de Pirituba, no interior da Bahia. Nela, o ramo desenvolveu uma oficina de Arduino para estudantes de uma das escolas da rede pública da cidade. A partir deste entusiasmo e pelo desejo de compartilhar conhecimento, criou-se a liga Programa de Professor em Serviço (em inglês, *Teacher In-Service Program*) – (TISP) – em nossa universidade.

Inicialmente, nossa liga contava com a participação de 11 voluntários. Atualmente, somos 19 membros voluntários, todos graduandos dos cursos de engenharia (Engenharia de Computação, Engenharia Civil e Engenharia de Alimentos) da própria universidade. O objetivo desta liga é apresentar as ciências percebidas nos respectivos cursos de graduação dos membros para a comunidade externa à universidade, a fim de fomentar o interesse do público assistido pelas intervenções em relação às áreas de atuação apresentadas nas atividades desenvolvidas pelo programa.

Ao longo de um ano e meio, nosso grupo realizou diversas atividades. Iniciamos com duas oficinas de programação em Python, sendo uma para alunos do ensino superior da própria universidade e a outra para alunos do ensino médio, em busca de atrair estudantes para a área. Em seguida, promovemos uma oficina de informática básica para estudantes recém-formados no Ensino Médio e, atualmente, estamos produzindo publicações nas redes sociais, além de vídeos com assuntos relacionados a tecnologia.

O presente artigo tem o propósito de relatar a nossa experiência e expor a percepção dos integrantes do TISP acerca do ensino da computação para a comunidade escolar e acadêmica de nossa região. Tais percepções surgiram através de atividades de ensino e monitoria para estudantes da educação básica e superior realizadas na forma de oficinas e minicursos. Deste modo, este artigo procura colaborar com a construção de experiências similares e encorajar práticas lideradas por estudantes no contexto da computação.

## 2 FUNDAMENTAÇÃO TEÓRICA

Apresentamos aqui o Programa de Professor em Serviço, os fundamentos pedagógicos da abordagem utilizada pelo grupo e alguns trabalhos correlatos.

### 2.1 Teacher in-Service Program (TISP)

Proposto pelo Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE), o Teacher In-Service Program (TISP) ou Programa de Professor em Serviço, tem como missão principal apoiar professores da educação básica no engajamento dos estudantes em relação a temas de tecnologia para que, no futuro, mais estudantes optem pela área da engenharia. Kam et al. apresentam os componentes principais do programa: (i) Desenvolver e testar planos de aula; (ii) Realizar oficinas de treinamento para ensinar voluntários em como se conectar e fazer parcerias com a comunidade pré-universitária utilizando estes planos de aula; (iii) fornecer treinamento aos professores que levarão os planos de aula para as salas de aula [24].

No ramo IEEE UEFS, as propostas originais tiveram que ser adequadas para o formato da educação brasileira. Os estudantes voluntários, em vez de fornecer o apoio aos professores, são agentes ativos que entram em contato diretamente com os estudantes do ensino médio e superior a partir de oficinas e minicursos que apresentam conceitos da engenharia a partir da metodologia da aprendizagem por pares.

### 2.2 Sociointeracionismo

Vygotsky explora, em seus estudos sociointeracionistas, que os contextos culturais e de interação social são importantes para a aprendizagem, explicando o quão relevantes são estes contextos para o desenvolvimento do estudante [52]. O enfoque dele, entretanto, é na formação das crianças.

Vygotsky também aborda, em alguns de seus trabalhos, o conceito de Zona de Desenvolvimento Proximal (ZDP). Ele a caracteriza como a distância entre o nível de desenvolvimento real, que se costuma determinar através da solução independente de problemas, e o nível de desenvolvimento potencial, determinado através da solução de problemas sob a orientação de um adulto ou em colaboração com pares mais capazes [52]. Este conceito pode ser aplicado em contextos universitários quando há atividades de monitoria, em que outros estudantes colaboram para a resolução de problemas e para ensinar conceitos, já que há interações com indivíduos mais experientes auxiliando os menos experientes a desenvolverem suas potencialidades [32, 37].

### 2.3 Aprendizagem por pares

A aprendizagem por pares é uma metodologia ativa de ensino-aprendizagem em que os estudantes são colocados como atores centrais do processo, atuando no papel de tutores para seus pares, sem necessariamente contar com a intervenção de um professor profissional [9, 50]. Um dos principais benefícios relacionados à aprendizagem por pares é que, ao separar materiais e preparar aulas, os estudantes tutores compreendem melhor os conteúdos que irão ensinar, já que que o processo cognitivo de estudar para ensinar é diferente do de estudar para uma prova. Os aprendizes são beneficiados pela habilidade dos tutores de ensinar o conteúdo de forma mais assertiva, além da sensação de proximidade que é

gerada já que as partes envolvidas estão no mesmo "nível", sendo todos estudantes [53]. Uma outra vantagem dessa metodologia, por ser centrada no estudante, é o desenvolvimento de habilidades intelectuais tais como a capacidade de síntese e a resolução de problemas [46].

## 2.4 Ferramentas de apoio às atividades do TISP

Arduino é uma plataforma de prototipagem eletrônica de código aberto para a criação de protótipos de software e hardware livres. É uma placa única, bastante simples de usar e, frequentemente, utilizada como ferramenta de suporte ao ensino de eletrônica [2, 10] e programação [17, 30]. Uma placa Arduino pode adquirir informação do ambiente externo através de seus pinos de Entrada/Saída (I/O<sup>1</sup>), analógicos ou digitais, quando conectados a módulos de sensores externos [16]. Além de captar dados, o Arduino ainda pode atuar no ambiente controlando componentes [36] como lâmpadas e atuadores. O Scratch para Arduino (S4A<sup>2</sup>) é uma variação da ferramenta de programação em blocos Scratch. S4A utiliza a linguagem de programação em blocos para a prototipagem de circuitos no Arduino, através do gerenciamento dos sensores ou atuadores conectados às portas da placa. Pereira and Bastos revelam a importância de aplicar uma oficina de robótica educacional com estudantes de 9º ano e extrair resultados através da compreensão e envolvimento dos alunos com a teoria e a prática, desenvolvendo habilidades como o raciocínio lógico, o trabalho em grupo e a disciplina na organização do trabalho, além de despertar o interesse dos estudantes para as áreas de tecnologia [34]. No trabalho de Barbero et al., os autores apresentam a contribuição preliminar para um currículo de primeiro ano de escolas técnicas do ensino médio [5]. Nele, os estudantes desenvolveram atividades de programação, onde Scratch é usado para programar e desenhar, e Arduino, para o controle de dispositivos.

Python é uma linguagem de programação simples e versátil, que vem ganhando adoção nos últimos anos [23, 41]. Ela combina os paradigmas procedural, funcional e orientado a objetos. Além de ser uma linguagem bastante utilizada por programadores profissionais, ela tem se destacado como primeira linguagem em cursos introdutórios de programação, tornando-se uma alternativa viável em comparação com linguagens mais tradicionais como C e Java [26]. O trabalho de Oldham destaca que os problemas enfrentados por estudantes iniciantes de Python não são piores do que os que enfrentariam caso se optasse por qualquer outra escolha para uma primeira linguagem [33]. Algumas das vantagens oferecidas por Python são: sintaxe simples e fácil, reforço de estrutura através de indentação, disponibilidade de bibliotecas, entre outras [1, 39].

*Google Apps* é um conjunto de ferramentas online do Google que permite o trabalho colaborativo online, ou seja, o desenvolvimento de atividades em grupo sem que os integrantes precisem estar no mesmo lugar ao mesmo tempo. Dentre as ferramentas, as mais conhecidas são o Google Documentos, o Apresentações e o Planilhas, que permitem a criação de arquivos similares ao *Microsoft Word*, ao *PowerPoint* e ao *Excel* [3]. Além da possibilidade de compartilhamento e trabalho colaborativo, outros diferenciais dessas ferramentas são: a mobilidade, é possível acessar os documentos

através de uma conta do Google em qualquer dispositivo; o versionamento, o controle de versão traz um histórico de versões desde a criação do documento até a última modificação, exibindo ainda o autor de cada uma das alterações; e, por fim, a sincronização, a possibilidade de trabalhar offline e sincronizar as alterações quando houver conexão disponível com a internet. Alguns estudos relatam a implementação de *Google Apps* e *Chromebooks* na educação como ferramentas de apoio à aprendizagem. Bartolo traz os resultados da integração de *Google Apps* no currículo de escolas públicas [6]. As experiências revelaram uma atitude positiva compartilhada e aumento da motivação dos estudantes. Já no ensino superior, o trabalho de Widodo [54] mostra como a disponibilização do material didático e a ferramenta de comunicação online para estudantes universitários deve ser uma prioridade, pois pode ser acessada a qualquer hora e em qualquer lugar. Assim, o uso e adoção dessas ferramentas podem ajudar a melhorar o ensino e a aprendizagem [4].

## 2.5 Diretrizes da Sociedade Brasileira de Computação

Os referenciais para a formação em computação na educação básica da Sociedade Brasileira de Computação (SBC) são descritos por Raabe et al [38]. Nele, são apresentados três eixos de conhecimentos da área de Computação, os quais servem como base fundamental para a criação das atividades do TISP: pensamento computacional, mundo digital e cultura digital.

O eixo **Pensamento Computacional**, busca resolver problemas, projetar sistemas e compreender o comportamento humano [35, 55], ou seja, utilizar conceitos da computação para atividades diárias ou profissionais, desenvolvendo as competências de abstração, automação e análise.

O eixo **Mundo Digital**, envolve a atual situação da sociedade, altamente conectada através da Internet, os seus pilares desenvolvem as competências de codificação, processamento e distribuição.

Por fim, o eixo **Cultura Digital**, envolve os impactos da computação na sociedade, trabalhando com as questões éticas e morais desenvolvendo as seguintes competências: computação e sociedade, fluência tecnológica e ética digital.

## 2.6 Trabalhos Relacionados

Kam et al. apresentam um projeto sobre a educação em engenharia, computação e tecnologias (em inglês, *Engineering, Computing and Technology (ECT) Education*) [24]. Neste artigo, apresentam ainda o TryEngineering.org, um portal web que se tornou popular entre a comunidade pré-acadêmica e acadêmica, e a relevância do TISP no fornecimento de informações confiáveis e claras sobre a ECT para professores, orientadores escolares, pais e alunos. O projeto do IEEE visou melhorar a compreensão das oportunidades oferecidas pela ECT, além de mostrar aos alunos a ECT de forma estimulante, de maneira a aumentar a quantidade de jovens a escolherem a engenharia como carreira. O portal web conta com planos de aula para professores, banco de dados de programas de ECT e jogos para alunos, e conseguiu atingir um número significativo de pessoas de muitos países. Além disso, o TISP organizou diversas oficinas que atingiram alguns países, entre os quais, Malásia, África do Sul e Estados Unidos, obtendo um bom feedback dos participantes.

<sup>1</sup> Input/Output

<sup>2</sup> Scratch for Arduino

No trabalho de da Cruz Alves et al., é apresentada a programação em Scratch como recurso de auxílio no ensino de História e Estudos Sociais no Ensino Fundamental de uma escola em Florianópolis/SC. Nessa experiência, os estudantes criaram jogos utilizando essa linguagem de programação, com isso eles desenvolveram seu conhecimento no ambiente Scratch e em conceitos de computação, como condicionais, laços, concorrência, interatividade e outros, assim como também teve o papel de reforçar os conhecimentos das disciplinas de História e Estudos Sociais [15].

Uma experiência realizada por Marques et al., com o intuito de introduzir jovens do ensino médio ao mundo da programação, utilizou-se da linguagem Python, por esta ser uma linguagem simples e poderosa, e sua biblioteca pyGame, voltada para o desenvolvimento de jogos com interface gráfica, para desenvolver jogos [27]. Ao final da oficina, foi possível observar que o uso de jogos serviu como um fator motivacional para os participantes, que houve um aumento no interesse dos alunos pela área de computação e que ficaram satisfeitos com os conteúdos abordados na oficina.

Roque expõe a criação e acompanhamento de um curso introdutório de programação, também na linguagem Python, para alunos do ensino médio [45]. O projeto foi realizado por um graduando e bolsista de extensão em conjunto com os gestores da Seara da Ciência, ambiente pertencente à Universidade Federal do Ceará, com o objetivo de oferecer aos estudantes um contato inicial com a lógica computacional e matemática, além de avaliar a disponibilidade dessa disciplina no âmbito escolar. Os resultados obtidos foram satisfatórios, mostrando que alunos podem assimilar essa temática desde o ensino médio. Pôde-se perceber ainda que o ensino da programação contribui para ajudar os alunos a enxergarem o universo que se esconde por trás das tecnologias.

Costa et al. apresenta dados de um curso de programação em Python desenvolvido por bolsistas de extensão do Laboratório de Informática para Educação (LipE), da Universidade Federal do Rio de Janeiro, com o objetivo de incentivar jovens do ensino médio a criarem interesse pela área da ciência da computação [14]. O artigo constatou que, apesar das dificuldades, os alunos têm interesse pelas áreas da computação. Para os bolsistas de extensão que participaram do curso como professores, foi uma experiência relevante para a sua formação profissional, pois puderam vivenciar o trabalho docente.

Saico et al. traz o uso do Scratch como ferramenta principal em um curso ministrado por estudantes do curso de Licenciatura em Computação da Universidade Federal da Paraíba para introduzir o pensamento computacional e algorítmico para os alunos do segundo e terceiro ano do ensino médio [47]. Teve como base uma proposta de estimular o desenvolvimento de pequenos softwares, tais como jogos e animações. Os resultados sugerem que os alunos se sentiram curiosos e interessados por programação, além de revelar forte aptidão de alguns alunos para a área. Ademais, os estudantes de Licenciatura em Computação tiveram um fortalecimento de sua formação, adquirindo conhecimento e experiência para sua formação profissional.

Na revisão sistemática, realizada por Zanetti et al., são respondidas questões acerca do ensino de programação na educação básica, visando o desenvolvimento do Pensamento Computacional. Nessa revisão que contou com 15 trabalhos realizados entre 2012 e 2015, apontaram o Scratch como uma das ferramentas mais utilizadas, revelando também o ensino médio/técnico como um dos públicos

mais abordados nas pesquisas. Sobre as habilidades de PC, foram destacadas a coleta, análise e representação de dados, decomposição de problemas, abstração e desenvolvimento de algoritmos [56].

Stewart apresenta uma análise da utilização de *Peer-led Team Learning*, um modelo de ensino colaborativo liderado por pares, em um curso de CS0. Nesta experiência, foi apresentada uma melhora no desempenho dos alunos e um avanço de habilidades de liderança dos tutores [49].

Roberts et al. mostram uma experiência de *Undergraduate Teaching Assistant (UTA)* na universidade de Stanford. Eles afirmam que é esta uma boa estratégia para dar suporte em disciplinas introdutórias de Ciência da Computação. Também apresentam os benefícios por parte dos alunos e dos *teaching assistants* [44]. Essa experiência inspirou a criação de um projeto semelhante na Universidade do Arizona, na Universidade de Washington e em outras [20, 42].

Gordon et al. apresentam uma visão do programa de UTA da Universidade da Virginia [22]. Neste programa, estudantes da universidade participaram como instrutores em um curso direcionado aos calouros. O curso em questão é interdisciplinar e centrado no aluno, a fim de desenvolver algumas habilidades como escrita, comunicação, pensamento crítico e afins, além de aumentar o engajamento dos alunos e prepará-los para a vida universitária. Nesse estudo, foram incluídos alguns relatos dos calouros participantes, que afirmaram se sentir mais confortáveis pelo fato de os assistentes enxergarem melhor o ponto de vista de estudante e por não serem muito mais velhos, o que denota uma sensação de proximidade provinda da aprendizagem por pares.

Em outro trabalho, Mirza et al. apresentam em uma revisão sistemática da literatura acerca dos programas de UTA, cujos benefícios relatados são os a formação de habilidades interpessoais, aumento na aprendizagem dos estudantes e satisfação de modo geral [31].

No artigo escrito por Berto et al., foi feito um projeto com o objetivo de introduzir o pensamento computacional por meio do ensino de programação com alunos do 5º ano de uma escola. As atividades iniciam-se com atividades desplugadas, que mostraram aos estudantes como o computador funciona e o que ele pode executar. Todos os conceitos de programação são introduzidos por atividades plugadas com o uso da linguagem Scratch na semana posterior, como instruções de direção, laços de repetição, variáveis e condicional. As semanas posteriores vão-se alternando as atividades desplugadas e plugadas, com o objetivo de identificar se o conteúdo que foi ensinado na semana anterior está sendo compreendido. A escola teve uma grande satisfação com os resultados positivos, pois ao final do curso pode-se perceber que os participantes demonstraram melhorias tanto na decomposição da solução de um problema de matemática, como também na organização sequencial de ideias. Também foi notada a baixa evasão dos alunos durante o projeto.

Matias et al. explicam o desenvolvimento de uma oficina dividida em 3 etapas, com a última sendo o ensino da linguagem python. Com o objetivo de aplicar um método que ofereça às crianças e jovens acesso à educação de modo inclusivo, um ambiente agradável onde alunos se comuniquem entre si e com os professores, que todos se interessem e participem independentemente da dificuldade. O modelo de ensino foi bem aceito e teve um resultado positivo no rendimento dos estudantes.

### 3 METODOLOGIA

Nesta seção, descrevemos o cenário e participantes do grupo, bem como os participantes das atividades, as ferramentas e o planejamento geral das experiências realizadas.

#### 3.1 Cenário

O TISP UEFS nasceu com a necessidade que vimos, como universitários e futuros engenheiros, de compartilhar as experiências aprendidas no decorrer das nossas graduações e, principalmente, os conhecimentos compartilhados entre os integrantes do ramo estudantil do IEEE de nossa instituição.

Já no primeiro mês de atuação, éramos 11 voluntários engajados em encontrar escolas e instituições interessadas em nossa contribuição e estabelecer relações de colaboração com educadores. A proposta, desde o início, era desenvolver atividades que aumentassem o nível de conhecimento técnico de estudantes pré-universitários, além de incentivá-los a seguirem carreiras voltadas às ciências e à engenharia. Com o passar do tempo, também voltamos nossos esforços para a difusão deste conhecimento no próprio meio acadêmico, estendendo a outros cursos que normalmente não têm acesso amplo a conteúdos de computação. Nestas atividades, descobrimos diversos estudantes com talentos para as ciências exatas e a programação. Cinco deles se interessaram em integrar o nosso grupo TISP. Atualmente, somos um grupo de 19 voluntários.

#### 3.2 Participantes

As atividades descritas neste trabalho foram desenvolvidas pelo TISP da nossa universidade e são divididas em três experiências: a Oficina de Introdução à Programação em Arduino, a Oficina de Informática Básica e o Minicurso de Aprendizagem de Programação em Python. Cada experiência foi aplicada uma ou mais vezes, por vezes em contextos diferentes. Traremos aqui um resumo dos participantes de cada experiência.

A Oficina de Introdução à Programação em Arduino consistiu de apenas uma experiência e foi mediada por cinco estudantes de graduação em Engenharia de Computação (membros/voluntários do IEEE UEFS), uma estudante de mestrado em Ciência da Computação e dois professores do Departamento de Tecnologia da universidade. Ela contou com a participação de 23 estudantes, todos do ensino médio público regular da cidade de Piritiba, Bahia, dos quais 14 (60,9%) eram do sexo feminino e 9 (39,1%), do sexo masculino.

O Minicurso de Aprendizagem de Programação em Python foi realizada em três turmas, em dois contextos diferentes. O primeiro contexto, em duas turmas, foi mediado por 10 voluntários e contou com a participação de 35 estudantes universitários de diversos cursos de graduação da Universidade Estadual de Feira de Santana, dos quais, 15 (42,9%) eram do sexo feminino e 20 (57,1%), do sexo masculino. O outro contexto, foi mediado por 5 voluntários e contou com a participação de 37 estudantes do Colégio Anísio Teixeira da rede privada de Feira de Santana, dos quais 20 (54,1%) eram do sexo feminino e 17 (45,9%), do sexo masculino.

A Oficina de Informática Básica foi realizada em três turmas, com estudantes diferentes. No total, ela foi mediada por 9 estudantes voluntários e contou com a participação de 46 estudantes e egressos da rede pública de ensino de Santo Estêvão, dos quais 33 (71,7%) eram do sexo feminino e 13 (28,3%) do sexo masculino.

### 3.3 Planejamento

A Oficina de Introdução à Programação em Arduino foi planejada com carga horária de oito horas, em um mesmo dia, distribuídas em dois turnos. Um dos estudantes de graduação se encarrega de ministrar as aulas e os demais o auxiliam, monitorando o desenvolvimento das atividades. Os professores apenas coordenam a oficina, deixando os estudantes livres no processo de integração com os participantes, contribuindo na aprendizagem em pares. O conteúdo é dividido em duas partes. Inicialmente, é apresentado o Arduino, suas especificações e componentes básicos, como os pinos analógicos e digitais e o sinal PWM. Posteriormente, é explanado sobre como os componentes elétricos, resistor, LED<sup>3</sup> e *Push Button* (botão) funcionam e como conectar esses dispositivos ao Arduino. Em um segundo momento, o foco do estudo é a programação, e consiste em apresentar os comandos do S4A e ensinar como enviar e receber dados da placa Arduino. Logo após, os estudantes devem iniciar as montagens em forma de desafios.

O Minicurso de Aprendizagem de Programação em Python foi planejado com 20 horas de duração, divididas em 10 aulas. Cada aula é ministrada por um tutor e os demais voluntários atuam como monitores. A Tabela 1 apresenta os conceitos trabalhados no minicurso:

Dia	Assunto
1	Lógica de Programação
2	Introdução à Linguagem Python
3	Estruturas de Decisão
4	Estruturas de Repetição
5	Funções
6	Funções Matemáticas e Strings
7	Estruturas de Dados I
8	Estruturas de Dados II
9	Arquivos I
10	Arquivos II

**Tabela 1: Conceitos trabalhados no minicurso de Programação com Python para o Ensino Superior**

A primeira aula tem um tom introdutório, na qual é apresentada a oficina e os dois conceitos que consideramos básicos para a aprendizagem de programação, independente de linguagem: Algoritmos e Lógica de Programação. Para o ensino de algoritmos, é utilizada a pseudolinguagem Portugol, um pseudocódigo em português estruturado. Enquanto que, para entender o fluxo de um programa e a lógica de programação, são utilizados fluxogramas. A partir da segunda aula, é introduzida a linguagem de programação Python, iniciando com o conceito de variáveis, tipos de dados, Entrada/Saída de dados e operações aritméticas. Na terceira aula, são incluídas as estruturas de decisão, junto aos operadores condicionais e lógicos. Na aula seguinte, estruturas de repetição, como *for* e *while*, são trabalhadas. Funções, parâmetros e retorno são incluídos na quinta aula, encerrando os conceitos mais fundamentais da programação. Da sexta aula em diante, são discutidos os seguintes conceitos: funções matemáticas e funções com Strings, estruturas de dados como

<sup>3</sup>Light Emitter Diode

tuplas, listas, conjuntos e dicionários e, finalizando, manipulação de arquivos. A ideia é que seja feito, na parte inicial de cada aula, um apanhado teórico dos conceitos a serem trabalhados. Para isso, os materiais produzidos são planejados para que as explicações seja breves e objetivas, mas eficientes. A outra porção da aula é destinada à prática, quando são apresentados problemas relacionados aos conteúdos da aula e os participantes devem tentar construir soluções com o mínimo de auxílio possível dos monitores.

Um minicurso de programação com Python de curta duração foi planejado para os estudantes do Ensino Médio de uma escola particular durante as aulas da disciplina de Física. A Tabela 2 apresenta os conceitos trabalhados.

Dia	Assunto
1	Lógica de Programação
2	Introdução à Linguagem Python
3	Estruturas de Decisão
4	Estruturas de Repetição
5	Avaliação

**Tabela 2: Conceitos trabalhados no Minicurso de Programação com Python para o Ensino Médio**

Diferentemente do curso voltado para o Ensino Superior, este apresentou, em seu último dia, uma avaliação que buscou relacionar o conteúdo aprendido pelos estudantes na programação com as aulas de Física.

A Oficina de Informática Básica foi planejada para quatro horas de duração, que optamos por realizar em um mesmo dia. Contamos com um tutor e os demais voluntários auxiliam com monitoria. O intuito desta oficina é tanto discutir conteúdos de informática e computadores em geral, como mostrar as vantagens de utilizar ferramentas colaborativas para a criação de documentos. Para isso, utilizamos as ferramentas do Google Apps.

## 4 NOSSA EXPERIÊNCIA

As experiências obtidas durante a nossa jornada no processo de ensino são relatadas nesta seção.

### 4.1 O Grupo

As reuniões presenciais ocorrem em um laboratório de pesquisa da universidade, cuja permissão de uso foi obtida com o professor responsável. Antes do seu início, dois integrantes recebem funções para uma melhor organização da discussão: o primeiro é o secretário de quadro, a pessoa irá organizar as ideias e questionamentos da reunião em um quadro visível a todos; o segundo é o secretário de mesa, que irá transcrever as informações do quadro na ferramenta *Trello*, que estarão disponíveis para todos acessarem a qualquer momento. É importante que essa transcrição esteja organizada para poder situar aqueles que não puderam comparecer à reunião.

Apesar de o grupo possuir papéis de liderança, ele possui uma estrutura de organização horizontal, na qual todos os integrantes possuem espaço para expressarem suas opiniões livremente e autonomia para tomarem decisões. Por exemplo, os membros podem decidir qual problemática querem atacar, qual assunto seria interessante tratar, como irão produzir o material que será utilizado,

entre outras questões. Todas as atividades são planejadas com a participação ativa de todos.

Após a divisão dos papéis, os presidentes do grupo lideram a reunião, sempre dando voz a todos os participantes. Inicialmente, são discutidas as atividades correntes, oficinas, planejamentos ou estudos, buscando sempre procurar destacar o que está dando certo ou o que precisa melhorar. Após esta discussão, passa-se a um momento de *brainstorming*, no qual os estudantes apresentam novas ideias, sempre ouvindo e respeitando a opinião dos outros.

A partir das ideias que são propostas pelos integrantes, as atividades que serão realizadas com o público são criadas. Em um primeiro momento, existe uma divisão de grupos: as pessoas que possuem mais afinidade com um determinado assunto são agrupadas e irão trabalhar em conjunto. O ideal para a organização de cada subgrupo é a atribuição de cargos de liderança e secretário de mesa: o primeiro irá liderar o seu subgrupo e reportar os acontecimentos para o grupo geral, e o segundo irá organizar as ideias e os questionamentos encontrados em um quadro do *Trello* para o seu subgrupo.

Para cada atividade, alguns passos devem ser seguidos antes de sua realização. O primeiro é a revisão da literatura: os estudantes irão pesquisar trabalhos relacionados a fim de obter uma base para criar os materiais que serão utilizados posteriormente. Após esta revisão, é necessário que os estudantes realizem uma capacitação interna, onde eles irão estudar em grupo o assunto que irá ser apresentado. Esta capacitação pode ocorrer presencial ou virtualmente, dependendo da disponibilidade dos seus integrantes e respeitando seus horários, pois eles possuem outras responsabilidades com a universidade. Após esta capacitação interna, os integrantes apresentam as propostas das aulas e o seu conteúdo para o grupo geral, onde irão receber *feedback*. Por fim, a atividade, que pode ser uma oficina, minicurso ou palestra, será divulgada nas redes sociais para alcançar o seu público.

As habilidades previamente mencionadas são mais trabalhadas durante esse ciclo de planejamento das atividades, já que os próprios estudantes do subgrupo serão responsáveis por realizar uma pesquisa sobre o assunto, procurando na literatura, com outros estudantes bolsistas de extensão ou iniciação científica, ou até com professores. Além disso, serão também responsáveis pela criação do material de divulgação da atividade, buscando um maior engajamento nas redes sociais ou a partir de cartazes que serão colados em murais da universidade ou das instituições parceiras. Finalmente, deverão procurar realizar parcerias com outros laboratórios e instituições que possuem a infraestrutura adequada para tornar possível a realização das atividades previamente elaboradas.

Para a realização das aulas, é realizado um rodízio entre os estudantes a fim de permitir que todos tenham a oportunidade de ensinar. Assim, o professor da vez vai ser o responsável por criar o material de sua aula e realizar a apresentação para os alunos. Os outros participantes que não irão ministrar a aula estarão presentes como monitores, auxiliando os alunos.

A inscrição para novos membros ocorre a cada início de semestre. Esta seleção ocorre através de uma entrevista realizada pelos presidentes do grupo, no qual buscam conhecer melhor os candidatos e suas habilidades previamente conhecidas. É importante ressaltar que a ausência de certas habilidades não é um fator eliminatório,

visto que o grupo tem como principal finalidade trabalhar essas habilidades.

A presença de novos membros é bastante importante para o grupo, pois, com mais pessoas, será possível realizar mais atividades. Sendo assim, buscamos ao máximo fazer com que todos os membros, novatos ou não, sintam-se acolhidos. O grupo possui momentos de descontração, onde os integrantes fortalecem os laços de amizade, o que é fundamental para a manutenção do grupo.

## 4.2 Oficina de Introdução à Programação em Arduino

Em dezembro de 2018, alguns estudantes do ramo estudantil IEEE UEFS foram convidados, em parceria com uma escola do município e a Prefeitura de Piritiba, a desenvolver uma atividade com estudantes do ensino médio da cidade. O intuito era realizar atividades voltadas à programação e à engenharia, para isto, foi planejada uma oficina de Arduino. Viajamos mais de 180 km para ministrá-la. Nela, foi utilizada uma abordagem lúdica, que instigasse a atenção e a motivação dos participantes, ao mesmo tempo que despertando o interesse deles pelo tema.

Na primeira etapa da oficina, foram abordados os conceitos principais de eletrônica necessários para manipular o Arduino. A segunda etapa consistiu da apresentação da interface do sistema S4A, os comandos e a comunicação com a placa. Por último, foi proposto o desenvolvimento de um projeto para que eles praticassem o conteúdo apresentado. O projeto consistia no desenvolvimento de um protótipo de piano de frutas utilizando bananas e componentes eletrônicos. Ao final do dia, os projetos que ficaram mais completos foram apresentados pelos estudantes desenvolvedores para os demais e professores, explicando tanto o código como a montagem.

Uma das principais características observadas pelos monitores é que os participantes se sentiam particularmente estimulados nas etapas em que a programação era trabalhada em conjunto com a manipulação do hardware. Também pudemos perceber que os estudantes tinham pouco ou nenhum conhecimento prévio de eletrônica básica e, apesar de apresentarem algumas dificuldades, se mantiveram bastante interessados ao longo da oficina. Ao final, com as apresentações, pudemos verificar que a maioria dos estudantes não tiveram dificuldades e conseguiram concluir seus projetos e explicá-los de maneira satisfatória. Um dos facilitadores apontou o fato de atividade ter sido realizada em dupla ou trio, instigando as discussões entre os estudantes.

## 4.3 Minicurso de Aprendizagem de Programação em Python

A primeira atividade realizada pelo grupo após a sua formalização em 2019, foi o minicurso de aprendizagem de programação em Python. Além da familiaridade dos integrantes do grupo com o tema de programação, visto que a maioria era do curso de Engenharia de Computação e já possuía um conhecimento prévio do assunto, o maior fator motivador para a escolha da linguagem foi a sua natureza simples e didática, se tornando ideal para o primeiro contato dos participantes com a programação. Foram realizados dois cursos de programação, um voltado para o ensino superior e outro, para o ensino médio.

**4.3.1 Ensino Superior.** Após o planejamento e a elaboração do material para o minicurso, a inscrição para os estudantes da UEFS foi aberta. Disponibilizamos duas turmas, para abranger um público maior e respeitar o limite de máquinas presentes no laboratório: uma na segunda-feira e outra na sexta-feira. Para o primeiro dia foram inscritos 27 estudantes e, no segundo, 32. No total, foram 10 dias de aula, com 2 horas de duração cada.

No **primeiro dia**, fizemos uma introdução sobre programação de computadores através da lógica de programação. Através de atividades práticas, ensinamos os conceitos de algoritmos, lógica de programação e fluxogramas. Notamos uma grande participação dos estudantes, que ficaram bastante entusiasmados quando pedimos para eles fazerem e posteriormente apresentarem um algoritmo em formato de pseudocódigo para representar algo presente no seu dia-a-dia como, por exemplo, passos para escovar os dentes, fazer compras, ir à universidade, entre outros.

Na **segunda aula**, fizemos uma introdução a Python, e os estudantes aprenderam a respeito das variáveis e os seus tipos, além da entrada e saída dos dados e operações aritméticas simples. Para possibilitar aos estudantes criarem os seus programas, utilizamos o serviço fornecido pelo *Repl.it*, onde os estudantes podem programar diretamente do navegador de Internet. Durante as aulas, a fim de praticar o que estava sendo ensinado, passamos pequenas atividades e, após 10 minutos, as corrigimos. Ao final da aula, uma atividade um pouco mais detalhada foi proposta para casa, para ser corrigida na aula seguinte.

O **terceiro dia** começou com a resolução da atividade proposta anteriormente, e verificamos que nem todos os estudantes a fizeram. Após a correção, apresentamos aos estudantes as estruturas de decisão. Trabalhamos as estruturas condicionais em Python, sempre fazendo analogias com pseudocódigo. Além disso, para poder trabalhar com as condições, apresentamos os operadores aritméticos e relacionais.

No **quarto dia**, apresentamos as estruturas de repetição *for* e *while*. Fazendo sempre correlações com os fluxogramas e o pseudocódigo, os estudantes conseguiram entender as ideias principais de loops através das pequenas atividades feitas em sala. Algumas vezes, foi necessária a ajuda de um monitor que estava presente. Porém, a maioria das dúvidas eram simples.

No **quinto dia**, apresentamos sobre modularização do código a partir das funções. Os estudantes aprenderam a diferença entre funções e procedimentos e como funciona o fluxo de execução do código quando existe uma chamada a uma função.

No **sexto dia**, os estudantes trabalharam com as funções matemáticas da biblioteca *math* de Python. Eles trabalharam com a radiação, hipotenusa, razões trigonométricas, entre outras. Após praticarem, os estudantes aprenderam operações com *strings*: conversão de uma *string* para um valor inteiro, obter o tamanho de uma *string*, entre outras.

Na **sétima e oitava aulas**, os estudantes aprenderam sobre as estruturas de dados em Python. Primeiro, apresentamos os conceitos de Listas e Tuplas, e, em seguida, conjuntos e dicionários. Percebemos que os estudantes tiveram dúvidas a respeito da diferença entre essas estruturas, além de como realizar a sua inicialização.

Nas **duas últimas aulas**, abordamos o tema de arquivos. Na primeira, apresentamos a motivação para a utilização de arquivos e como realizar operações de leitura e escrita de arquivos. Já na

segunda, recapitulamos a aula anterior e ensinamos como utilizar o comando *with* além de alguns métodos auxiliares como o *split()*.

Antes de finalizar a última aula, passamos um questionário a fim de verificar as percepções dos estudantes sobre a oficina, onde eles puderam indicar pontos de melhora. As respostas foram bastante positivas, que indicaram que a metodologia utilizada na oficina tornou o processo de ensino-aprendizagem descontraído e divertido, facilitando o aprendizado. Na primeira turma, concluíram o curso seis estudantes e, na segunda, oito. Alguns estudantes que participaram dessa oficina se interessaram pelo grupo TISP e, no início do semestre, entraram como voluntários e estão agora planejando conosco novas atividades para a comunidade.

Para alguns dos voluntários, aquela foi a primeira experiência de ensino, o que trouxe consigo o nervosismo e a ansiedade. Porém, o integrante que estava ministrando a aula sempre tinha o apoio dos outros colegas do grupo que estavam presentes como monitores.

O curso teve um público de cursos variados dentro da universidade: Engenharia de Computação, Engenharia Civil, Engenharia de Alimentos, Enfermagem, História e Economia. Após a realização das atividades, conforme a presença, os estudantes receberam um certificado de participação no minicurso.

**4.3.2 Ensino Médio.** Durante a realização do minicurso para o Ensino Superior, realizamos a mesma oficina de forma simplificada, agora voltada para os estudantes do segundo ano do Ensino Médio de uma escola particular. A oficina aconteceu durante as aulas de Física e o professor de Física computou dois pontos extra das pequenas atividades passadas para casa e de uma avaliação final. No total, 37 estudantes participaram da oficina.

A **primeira aula** apresentou aos estudantes a lógica de programação. A partir de atividades desplugadas, que são atividades onde são aprendidos conceitos de computação sem utilizar um computador, os estudantes praticaram os conceitos da lógica de programação e pensamento computacional.

Na **segunda aula**, os estudantes foram para o laboratório da escola, que possuía as ferramentas adequadas para a aprendizagem de programação com Python. Os estudantes apresentaram um bom desempenho nas atividades da aula, pois o conteúdo apresentado foi bastante simples (entrada e saída de dados e operações aritméticas). Antes do final da aula, a atividade para casa foi passada e os estudantes deveriam entregá-la através da plataforma *Google Classroom*.

Na **terceira aula**, a atividade para casa foi corrigida, os estudantes não tiveram grandes dificuldades com o código. Após a correção, estudantes trabalharam com as estruturas de decisão *if*. Assim como no ensino superior, fizemos correlações com o pseudocódigo a fim de facilitar o processo de aprendizagem dos estudantes. No final, outra atividade para casa foi proposta.

Na **quarta aula**, a atividade anterior foi corrigida, e notamos que os estudantes tiveram algumas dificuldades com o bloco condicional. Muitos esqueciam os dois pontos do bloco, o que é um erro comum em Python, além de não utilizarem uma lógica correta para a solução dos problemas. Após a correção das atividades, apresentamos somente a estrutura de repetição *for*. Ao final, propusemos outra atividade para casa.

Por fim, na **última aula**, antes da avaliação, a atividade para casa foi corrigida. Novamente encontramos erros de sintaxe. A

avaliação que passamos para os estudantes envolvia os assuntos que estavam sendo trabalhados na disciplina de física: magnetismo e ondas. Sendo assim, a avaliação contou com três questões e os alunos resolveram a prova em dupla e com consulta a materiais.

O professor de Física, também, foi um fator motivacional na oficina. Ele estava presente em todas as aulas, participando ativamente e realizando interações com os estudantes a fim de auxiliá-los. Por entender a importância da computação na sociedade e a interdisciplinaridade que existe entre as várias áreas do conhecimento, foi possível realizar a oficina com sucesso, obtendo bons resultados dos estudantes nas atividades e na avaliação final. Além disso, o feedback dos estudantes a respeito do formato da oficina foi bastante positivo.

#### 4.4 Oficina de Informática Básica

Ao final do ano, um dos vereadores da cidade de Santa Bárbara, localizada no interior do estado, entrou em contato conosco por meio das redes sociais propondo uma oficina de informática básica para estudantes e egressos do Ensino Médio da rede pública da cidade. Apesar de nosso foco estar em atividades centradas em programação ou hardware, entendemos que para dar este passo, às vezes é necessário fornecer uma formação básica preliminar para os estudantes, visto que muitos não sabem nem mesmo como manipular um computador.

A oficina foi realizada no Centro Juvenil de Ciência e Cultura - CJCC Feira, após formalizarmos uma parceria com os diretores da instituição. A escola possui suporte para realização da oficina e ofereceu um laboratório com computadores.

A Oficina de Informática Básica foi dividida em dois arcos principais. O primeiro, com conteúdos de informática como: o que é, para quê serve e como funciona um computador e seus componentes internos e externos; o sistema operacional Windows, como funciona a manipulação de arquivos neste sistema e atalhos existentes e, por fim, como funciona a Internet, o papel dos navegadores e dos serviços de e-mail. O segundo arco consiste na introdução e aplicação das ferramentas Google Apps, em especial, Google Documentos e Apresentações. Ao final da oficina, os estudantes foram divididos em grupos e fizeram uma apresentação de slides sobre um tema livre.

#### 4.5 Atividades durante a pandemia do COVID-19

Durante a pandemia do COVID-19, as atividades presenciais na universidade foram paralisadas. Sendo assim, tivemos que nos adaptar à nova realidade virtual, começando pelo maior uso de redes sociais, onde planejamos sequências de postagens informativas para a população. As postagens acontecem três vezes durante a semana em um horário pré-definido a fim de obter um maior engajamento dos seguidores.

Além das redes sociais, o grupo começou a utilizar a plataforma de vídeos *Youtube*<sup>4</sup> para disponibilizar vídeo-aulas pré-gravadas e editadas. Uma sequência de vídeos a respeito dos aplicativos do Google (Google Drive, Planilhas, Agenda e Docs) já foi postada. Somados todos os vídeos, já possuem mais de 300 visualizações.

<sup>4</sup>Disponível em <https://www.youtube.com/channel/UCkm9CsLopD4P7ZEPxYNPQmA>

Além dos vídeos pré-gravados, uma apresentação ao vivo foi realizada apresentando um minicurso de fundamentos do *LaTeX* com conceitos e práticas.

## 5 LIÇÕES APRENDIDAS

A partir da nossa experiência apresentada na sessão anterior, algumas lições podem ser destacadas:

**Pensamento computacional como competência transversal.** As oficinas, sejam de programação ou eletrônica, contribuem na construção do pensamento computacional pelos estudantes e proporcionam ferramentas para a solução de problemas através deste. Acreditamos que os conceitos trabalhados e as práticas realizadas têm sua importância para além da computação, revelando o desenvolvimento de habilidades muito importantes nos mais diversos âmbitos da vida dos estudantes.

**Independência no processo de articulação com a comunidade.** Os estudantes do grupo se organizaram de maneira independente para procurar laboratórios e instituições, e criar parcerias, a fim de poderem realizar as atividades voltadas à comunidade.

**Estrutura horizontal de liderança.** Não há uma decisão tomada unicamente pelos líderes, nem eles possuem a "palavra final". Existem reuniões de planejamento e todos os participantes possuem oportunidades para apresentar suas ideias e opiniões, e contribuir na tomada de decisões. Este foi um dos pontos mais apontados como positivos pelos membros.

**Oficinas mutáveis.** A partir do desenvolvimento da turma, o planejamento e a metodologia aplicados em uma oficina podem ser modificados a fim de favorecer o processo de aprendizagem dos estudantes. Por isso, é importante sempre buscar feedback dos participantes, visando aprimorar estes pontos durante o curso da oficina.

**Responsabilidades são motivadas e não atribuídas.** Nas reuniões de planejamento, os voluntários dão ideias e propõem as suas atividades de interesse. Os papéis (organização, desenvolvimento) são assumidos e não delegados. Desta forma, os estudantes se sentem motivados a estudar sobre o tema escolhido e cumprem os prazos com responsabilidade.

**Evasão de voluntários é desmotivadora.** Apesar das poucas ocorrências, o problema da evasão de voluntários do grupo é um fator preocupante. Algumas soluções encontradas foram realizar seleções para entradas periódicas de voluntários no grupo e proporcionar uma experiência de grupo interessante e divertida, para que os membros não fiquem desmotivados.

**Evasão de alunos é desmotivadora para o grupo.** Percebemos que, nas oficinas de programação com Python para o Ensino Superior, houve grande evasão por parte dos estudantes, o que pode ser desmotivador para os membros do grupo TISP. Uma maior aproximação entre os voluntários e os alunos, além da disponibilidade para tirar dúvidas extra-classe pode auxiliar na solução deste problema.

## 6 CONCLUSÕES

Neste trabalho, descrevemos as atividades de estudantes ensinando computação para a comunidade através do grupo TISP UEFS. Iniciamos descrevendo a formação do grupo em nossa universidade, com o objetivo de contribuir com o desenvolvimento de habilidades

importantes para os estudantes do século XXI. Descrevemos as atividades realizadas pelo grupo, realçando as experiências específicas de acordo com o tema abordado em cada uma delas. Realizamos uma oficina de Introdução à Programação com Arduino, para estudantes da rede pública de *Piritiba - BA*; três minicursos de Aprendizagem de Programação em Python, um para estudantes pré-universitários da rede privada de *Feira de Santana - BA* e dois para estudantes universitários de diversos cursos da nossa instituição, e, por fim, três oficinas de Informática Básica para estudantes e egressos da rede pública de *Santa Bárbara - BA*.

Nossas percepções iniciais revelam que a autonomia dos estudantes membros do grupo no processo de articulação com instituições externas e com a comunidade proporciona uma maior independência do grupo, tanto em relação ao ramo estudantil como à universidade. Um ponto positivo levantado pelos voluntários do grupo foi a liberdade para propor ideias e assumir a liderança de atividades de seus interesses dentro do grupo. Além disso, as atividades não são atribuídas pelos líderes e, sim, discutidas entre todos os membros do grupo, a fim de motivá-los a escolher e não obrigá-los a desenvolver uma determinada tarefa. Outra lição aprendida é que, apesar de o planejamento de uma oficina ser essencial para que todos os conteúdos e práticas sejam trabalhados adequadamente, ele não deve ser imutável, e vários aspectos da experiência podem ser moldados de acordo com o feedback dos participantes nas atividades, tais como o ponto de partida e o ritmo de ensino.

O maior problema identificado foi a questão da evasão, tanto dos participantes durante uma experiência, como dos próprios voluntários do grupo, e ambos são fatores desmotivadores. Foram propostas soluções que visam melhorar a dinâmica pessoal, tanto entre voluntários como entre voluntários e participantes, e dos processos do grupo.

Diversos caminhos podem ser trilhados de maneira não só a dar continuidade, mas expandir as atividades do grupo. Pretendemos continuar com a realização de oficinas de programação, eletrônica e informática, assim como novos temas que sejam importantes no desenvolvimento de habilidades do século XXI. Como trabalhos futuros, pretendemos desenvolver uma análise aprofundada a respeito dos impactos das experiências realizadas nas habilidades dos estudantes, como raciocínio lógico, resolução de problemas, entre outras. Consideramos que o trabalho aqui apresentado é um ponto de partida útil para que outros grupos educacionais, sejam parte ou não de um ramo estudantil IEEE, possam ser criados e consigam atingir positivamente a comunidade do seu entorno.

## AGRADECIMENTOS

Agradecemos aos participantes de todas as atividades realizadas, assim como aos demais membros do IEEE TISP UEFS. Agradecemos ainda aos professores e funcionários da UEFS, especialmente do NUCAE, e demais parceiros institucionais que ajudaram a viabilizar a realização das oficinas com a infraestrutura disponibilizada.

## REFERÊNCIAS

- [1] Krishna K Agarwal and Achla Agarwal. 2005. Python for CS1, CS2 and beyond. *Journal of Computing Sciences in Colleges* 20, 4 (2005), 262–270.
- [2] Rafael Machado Alves, Armando Luiz Costa da Silva, Marcos de Castro Pinto, Fabio Ferrentini Sampaio, and Marcos da Fonseca Elia. 2013. Uso do hardware livre Arduino em ambientes de ensino-aprendizagem. *Jornada de Atualização em Informática na Educação* 1, 1 (2013), 162–187.

- [3] Ramirose Attebury, Julie George, Cindy Judd, and Brad Marcum. 2013. Google docs: a review. *Against the Grain* 20, 2 (2013), 9.
- [4] Lawrence J Awuah. 2015. Supporting 21st-Century Teaching and Learning: The Role of Google Apps for Education (GAPE). *Journal of Instructional Research* 4 (2015), 12–22.
- [5] Alberto Barbero, Barbara Demo, Francesco Vaschetto, et al. 2011. A contribution to the discussion on informatics and robotics in secondary schools. *Proceedings RIE* (2011).
- [6] Paula Bartolo. 2017. Integrating google apps and google chromebooks into the core curriculum: A phenomenological study of the lived experience of public school teachers. (2017).
- [7] Washington Batista, Henderson Chalegre, João Paulo Sena, Allen Santos, David dos Santos, Carlos Rodrigues, and Roberto Bittencourt. 2015. Oficinas de aprendizagem de programação em uma escola pública através do ambiente scratch. In *Anais do XXIII Workshop sobre Educação em Computação*. SBC, 306–315.
- [8] Leticia Berto, Luciana Zaina, and Tiemi Sakata. 2019. Metodologia Para Ensino do Pensamento Computacional para Crianças Baseada na Alternância de Atividades Plugadas e Desplugadas. *Revista Brasileira de Informática na Educação* 27 (10 2019), 01. <https://doi.org/10.5753/rbie.2019.27.02.01>
- [9] David Boud, Ruth Cohen, and Jane Sampson. 1999. Peer learning and assessment. *Assessment & evaluation in higher education* 24, 4 (1999), 413–426.
- [10] Lucas Mellos Carlos, João Paulo de Lima, José Pedro Schardosim Simão, and Juarez Silva. 2016. block. ino: Um experimento remoto para ensino de lógica de programação, robótica e eletrônica básica. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, Vol. 5. 151.
- [11] Eduardo OC Chaves. 1998. Tecnologia e educação: o futuro da escola na sociedade da informação. *Campinas: Mindware Editora* (1998).
- [12] Eduardo OC Chaves. 2012. Tecnologia na educação, ensino a distância, e aprendizagem mediada pela tecnologia. *Revista de Educação PUC-Campinas* 7 (2012).
- [13] Leticia Saraiva Chaves, Carla Ilane Moreira Bezerra, et al. 2019. Ensino de Programação em Escolas Públicas: Relato de uma Ação do PET-TI. In *Anais do Workshop de Informática na Escola*, Vol. 25. 667.
- [14] Ana Cristina Costa, Ricardo Jullian, Cláudia Valéria Assis Mota, Alice Alves Franco, Victor Hugo Jardim Muniz, Luccas de Lima Maia, and Tiago Melo Liese. 2017. Python: Será que é possível numa escola pública de Ensino Médio?. In *Anais do Workshop de Informática na Escola*, Vol. 23. 255.
- [15] Nathalia da Cruz Alves, Christiane Gresse von Wangenheim, Pedro Eurico Rodrigues, Jean Carlo Rossa Hauck, and Adriano Ferreti Borgatto. 2016. Ensino de computação de forma multidisciplinar em disciplinas de história no ensino fundamental—um estudo de caso. *Revista Brasileira de Informática na Educação* 24, 3 (2016), 31.
- [16] Fábio Saraiva da Rocha, Guilherme Frederico Maranghello, and Márcia Maria Lucchese. 2014. Acelerômetro eletrônico e a placa arduino para ensino de física em tempo real. *Caderno Brasileiro de Ensino de Física* 31, 1 (2014), 98–123.
- [17] Luciano Frontino de Medeiros and Luana Priscila Wunsch. 2019. Ensino de programação em robótica com Arduino para alunos do ensino fundamental: relato de experiência. *Revista Espaço Pedagógico* 26, 2 (2019), 456–480.
- [18] Cassia de Oliveira Fernandez, Leandro Coletto Biazon, Alexandre AG Martinazzo, Irene Karaguilla Ficheman, and Roseli de Deus Lopes. 2015. Uma proposta baseada em projetos para oficinas de Internet das Coisas com Arduino voltadas a estudantes do Ensino Médio. *RENOTE-Revista Novas Tecnologias na Educação* 13, 2 (2015).
- [19] Beatriz Silva de Santana and Claudia Pinto Pereira. 2019. Aproximação de alunas do ensino básico do pensamento computacional: relato de experiência de uma oficina de eletrônica. In *Anais do Workshop de Informática na Escola*, Vol. 25. 69.
- [20] Jeffrey Forbes, David J Malan, Heather Pon-Barry, Stuart Reges, and Mehran Sahami. 2017. Scaling introductory courses using undergraduate teaching assistants. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on computer science education*. 657–658.
- [21] Selma Simonstein Fuentes. 2014. O porquê e o como das ciências na educação infantil. *Pátio – Ciências na educação infantil* 33 (2014), 08–11.
- [22] Jessica Gordon, Peter Henry, and Michaux Dempster. 2013. Undergraduate Teaching Assistants: A Learner-Centered Model for Enhancing Student Engagement in the First-Year Experience. *International Journal of Teaching and Learning in Higher Education* 25, 1 (2013), 103–109.
- [23] Joel Grus. 2019. *Data Science do zero: Primeiras regras com o Python*. Alta Books.
- [24] Moshe Kam, Yvonne Pelham, and Douglas Gorham. 2007. IEEE Activities in Pre-University Engineering, Computing and Technology (ECT) Education. In *2007 IEEE Meeting the Growing Demand for Engineers and Their Educators – 2010-2020 International Summit*, Vol. 50. IEEE, 1–6.
- [25] Cátia Mesquita Brasil Khouri, Gidevaldo Novais dos Santos, and Maria Silva Santos Barbosa. 2020. Mapeamento Sistemático em Metodologias de Ensino-aprendizagem de Programação. *Revista de Ciência da Computação* 2 (maio 2020), 13–27. <https://doi.org/10.22481/recic.v2i1.6669>
- [26] Greg Lindstrom. 2005. Programming with python. *IT Professional Magazine* 7, 5 (2005), 10.
- [27] Diego Lopes Marques, Luís Felipe Silva Costa, Max André de Azevedo Silva, and Ayla Débora Dantas S Rebouças. 2011. Atraindo alunos do ensino médio para a computação: Uma Experiência Prática de Introdução à Programação utilizando Jogos e Python. In *Anais do Workshop de Informática na Escola*, Vol. 1. 1138–1147.
- [28] Veronica Matias, Ewerton Monteiro, Érica Nogueira, and Carlo Emmanuel de Oliveira. 2016. SuperPython: experimento de ensino de programação para crianças. 647. <https://doi.org/10.5753/cbie.wcbie.2016.647>
- [29] Geisiane Souza Matos, Sara Silva, Fabiola Araújo, Yomara Pires, and Marcos Seruffo. 2019. Desenvolvimento de Jogos para Incentivar Meninas na Área da Tecnologia de Informação: um estudo de caso no ensino médio de escola pública. In *Anais do Workshop de Informática na Escola*, Vol. 25. 715.
- [30] Francisco Édson Nogueira Mélo, Raimundo Ricardo Matos da Cunha, Dyonad Renan Scolari, and Jhonatan Luiz Campos. 2011. Do Scratch ao Arduino: Uma proposta para o ensino introdutório de programação para cursos superiores de tecnologia. In *XXXIX Congresso Brasileiro de Educação em Engenharia*. 10.
- [31] Diba Mirza, Phillip T Conrad, Christian Lloyd, Ziad Matni, and Arthur Gatin. 2019. Undergraduate Teaching Assistants in Computer Science: A Systematic Literature Review. In *Proceedings of the 2019 ACM Conference on International Computing Education Research*. 31–40.
- [32] Elisete Gomes Natário and Acácia Aparecida Angeli dos Santos. 2010. Programa de monitores para o ensino superior. *Estudos de Psicologia (Campinas)* 27, 3 (2010), 355–364.
- [33] Joseph D Oldham. 2005. What happens after Python in CS1? *Journal of computing sciences in colleges* 20, 6 (2005), 7–13.
- [34] Alex Miguel Boaes Pereira and Thomas Jorge Moreira Bastos. 2019. Concepção e implementação da robótica educacional utilizando arduino e linguagem de programação introdutória scratch como ferramentas didáticas. (2019).
- [35] Francisco Tito Silva Santos Pereira, Luis Gustavo Araújo, and Roberto Bittencourt. 2019. Intervenções de Pensamento Computacional na Educação Básica através de Computação Desplugada. In *Anais do Workshop de Informática na Escola*, Vol. 25. 315–324.
- [36] Anderson Luiz Fernandes Perez, Renan Rocha Darós, Fernando Emilio Puntel, and Sandra Regina Vargas. 2013. Uso da Plataforma Arduino para o Ensino e o Aprendizado de Robótica. In *International Conference on Interactive Computer aided Blended Learning*.
- [37] Petrilson Pinheiro. 2018. Produção textual em contexto de ensino superior: rediscutindo perspectivas e procedimentos de ensino-aprendizagem. *Alfa: Revista de Linguística (São José do Rio Preto)* 62, 2 (2018), 325–343.
- [38] André Luis Alice Raabe, Avelino Francisco Zorzo, Ismar Frango, L Ribeiro, LZ Granville, L Salgado, M Cruz, N Bigolin, S Cavalheiro, and S Fortes. 2017. Referenciais de formação em computação: Educação básica. *Sociedade Brasileira de Computação* (2017).
- [39] Luciano Ramalho. 2015. *Python Fluente: Programação clara, concisa e eficaz*. Novatec Editora.
- [40] Felliipe Ramos and Lilian da Silva Teixeira. 2015. Significação da aprendizagem através do pensamento computacional no ensino médio: uma experiência com scratch. In *Anais do Workshop de Informática na Escola*, Vol. 21. 217.
- [41] Sebastian Raschka, Joshua Patterson, and Corey Nolet. 2020. Machine Learning in Python: Main developments and technology trends in data science, machine learning, and artificial intelligence. *Information* 11, 4 (2020), 193.
- [42] Stuart Reges. 2003. Using undergraduates as teaching assistants at a state university. *ACM SIGCSE Bulletin* 35, 1 (2003), 103–107.
- [43] Mitchel Resnick. 2007. All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten. *Creativity and Cognition* 2007, CC2007 - Seeding Creativity: Tools, Media, and Environments, 1–6. <https://doi.org/10.1145/1254960.1254961>
- [44] Eric Roberts, John Lilly, and Bryan Rollins. 1995. Using undergraduates as teaching assistants in introductory programming courses: an update on the Stanford experience. In *Proceedings of the twenty-sixth SIGCSE technical symposium on Computer science education*. 48–52.
- [45] Matheus Maia Roque. 2017. Curso Introdutório de Programação para Alunos do Ensino Médio. *Extensão em Ação* 1, 13 (2017), 31–43.
- [46] Lois Rubin and Catherine Hebert. 1998. Model for active learning: Collaborative peer teaching. *College Teaching* 46, 1 (1998), 26–30.
- [47] Pasqueline Dantas Scaico, Anderson Alves de Lima, Jefferson Barbosa Belo da Silva, Silvia Azevedo, Luiz Fernando Paiva, Ewerton Henning Souto Raposo, Yugo Alencar, and João Paulo Mendes. 2012. Programação no ensino médio: uma abordagem de ensino orientado ao design com Scratch. In *Anais do Workshop de Informática na Escola*, Vol. 1.
- [48] A Schleicher. 2018. É preciso continuar avançando. *Pátio-. Conhecimento científico no ensino médio, Porto Alegre. Ano IV* 12 (2018), 14–16.
- [49] Vivien Stewart. 2011. Improving teacher quality around the world. *Phi Delta Kappan* 92, 8 (2011), 93.
- [50] Keith J Topping. 2005. Trends in peer learning. *Educational psychology* 25, 6 (2005), 631–645.
- [51] Bernie Trilling and Charles Fadel. 2009. *21st century skills: Learning for life in our times*. John Wiley & Sons.
- [52] Lev Semenovich Vygotsky. 1980. *Mind in society: The development of higher psychological processes*. Harvard university press.

- [53] Neal A Whitman and Jonathan D Fife. 1988. *Peer Teaching: To Teach Is To Learn Twice*. ASHE-ERIC Higher Education Report No. 4, 1988. ERIC.
- [54] S Widodo. 2017. Implementing google apps for education as learning management system in math education. In *Journal of Physics: Conference Series*, Vol. 895. 0–5.
- [55] Jeannette M Wing. 2006. Computational thinking. *Commun. ACM* 49, 3 (2006), 33–35.
- [56] Humberto Zanetti, Marcos Borges, and Ivan Ricarte. 2016. Pensamento computacional no ensino de programação: Uma revisão sistemática da literatura brasileira. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Vol. 27. 21.