

Aplicação de um Método para Ensino de Programação Orientada a Objetos por meio de Aprendizagem Significativa e Computação Física

Humberto A. P. Zanetti¹, Marcos A. F. Borges¹, Ivan L. M. Ricarte¹

¹Faculdade de Tecnologia – Universidade Estadual de Campinas (UNICAMP) – Limeira – SP – Brasil

h016304@dac.unicamp.br, {marcosborges, ricarte}@ft.unicamp.br

Abstract. *This paper presents an experience report and its results about an Object-Oriented Programming workshop applying a method that uses premises of Meaningful Learning and Physical Computing, ComFAPOO (Physical Computing for Learning Object-Oriented Programming). This methodology has a teaching proposal aiming to minimize difficulties of novice students in the object-oriented paradigm. Results show that the method helped students to understand fundamental concepts of Object-Oriented Programming.*

Resumo. *Este artigo apresenta um relato de experiência e seus resultados sobre uma oficina de ensino de Programação Orientado a Objetos por meio de um método que utiliza premissas da Aprendizagem Significativa e Computação Física, ComFAPOO (Computação Física para Aprendizagem de Programação Orientada a Objetos). Este método propõe uma estratégia de ensino que busca amenizar as principais dificuldades de alunos iniciantes no ensino paradigma orientado a objetos. Os resultados de sua aplicação mostram que o método auxiliou os alunos participantes na compreensão de conceitos fundamentais da Programação Orientado a Objetos.*

1. Introdução

O ensino de Programação Orientada a Objetos (POO) ainda é um desafio, sendo foco de pesquisas recentes, principalmente sobre as dificuldades que são encontradas no processo [Gutiérrez, Guerrero e López-Ospina 2022; Abassi et al. 2021]. Desde o início do ensino desse paradigma, pesquisadores já relatavam problemas encontrados por seus alunos, como aponta o trabalho de Knudsen e Madsen (1988), demonstrando que ensinar POO é mais do que ensinar linguagens e as técnicas de programação, mas sim ensinar seus conceitos fundamentais. Beck e Cunningham (1989) apresentam uma reflexão sobre o ensino de programação e a dificuldade de seus alunos em migrar de um paradigma procedural para um com maior nível de abstração, como o paradigma POO.

O professor encontra seu primeiro desafio na criação de uma proposta de um currículo que possa abordar os fundamentos do paradigma POO. Bennedsen e Schulte (2007) fazem uma análise através de entrevistas com diversos professores e pesquisadores ao redor do mundo, não apresentando uma concepção única sobre o que é o ensino básico de POO, mas identificando como consenso que, no início do

aprendizado, o aluno deve entender como usar o “objeto” antes de implementar qualquer tipo de técnica.

O método ComFAPOO (Computação Física para Aprendizagem de Programação Orientada a Objetos) foi criada com o objetivo de apoiar os alunos a enfrentar essas dificuldades no processo de aprendizado de POO [Zanetti, Borges e Ricarte 2021]. Este artigo apresenta o relato de experiência na aplicação de oficinas de ensino de POO usando o método ComFAPOO, com 14 alunos de um curso técnico em Informática, com o objetivo avaliar seus efeitos e sua viabilidade como instrumento de ensino. A Seção 2 apresenta uma revisão bibliográfica sobre as dificuldades encontradas no ensino de POO. A Seção 3 apresenta as referências utilizadas para definir os fundamentos de POO que foram apresentados durante a oficina. Na Seção 4 é apresentada uma breve descrição do método. A Seção 5 descreve o perfil dos alunos e a dinâmica de aplicação da oficina. A Seção 6 apresenta os resultados e discussões sobre eles. Por fim, a Seção 7 apresenta as considerações finais.

2. Dificuldades encontradas no ensino de Programação Orientada a Objetos

Há muitos desafios associados ao ensino de POO. Gutiérrez, Guerrero e López-Ospina (2020) mostram que as dificuldades dos alunos vão desde a aplicação de conceitos básicos de programação, a relação entre classe e objeto e o desenvolvimento abstrato, até conceitos mais avançados, como não compreender a aplicação de herança, polimorfismo e reuso de código. Xinogalos (2015) reforça que as dificuldades primárias se concentram em compreender os conceitos de classe e objeto, fazendo com que o início do processo de aprendizagem seja desmotivador. Liberman, Beerli e Kolikant (2011) apresentam uma análise sobre as dificuldades dos alunos em compreender herança e polimorfismo, ressaltando que o principal problema é a falta de relação da aplicação desses conceitos em um cenário real.

Assim como em qualquer processo de aprendizagem, deficiências encontradas logo no início da aprendizagem de POO pode levar a uma cadeia de novos problemas de aprendizado, sendo intensificado a cada conceito novo apresentado. Logo nos conceitos iniciais de POO, o aluno já é estimulado a utilizar a habilidade de abstração e, ao mesmo tempo, representar um modelo em código. Lian, Varoy e Giacaman (2022) e Abbasi et al. (2021) apresentam, em suas pesquisas, uma listagem de equívocos e dificuldades comuns em cursos de POO, havendo ocorrências tanto com conceitos mais básicos quanto com os mais avançados.

Bennedsen e Caspersen (2006) mostram que mesmo alunos que possuem uma habilidade de abstração alta (observada por avaliações matemáticas) não apresentam uma habilidade de programação de mesmo nível. É apontado que, muitas vezes, o curso de programação tem o foco muito no instrumento (linguagem) e pouco no projeto (conceitos). Möstrom et al. (2008) argumentam que o problema para os alunos não é a capacidade de abstração, mas sim, a contextualização desses modelos e objetos em um cenário que faça sentido.

Conceitos abstratos podem ser melhor absorvidos se aplicados em cenários realistas ou envolverem objetos reais, que possam trazer interação e significado. O uso de ferramentas instrucionais que não sejam apenas baseadas em código-fonte pode amenizar a compreensão dos conceitos até mais avançados. Em seu trabalho, Möstrom et al. (2008) utilizaram objetos concretos como base para o exercício de abstração de

conceitos de Computação. Trory, Howland e Good (2018) demonstram que o uso de objetos concretos, que podem ser palpáveis e interativos, auxiliam no exercício de abstração de conceitos matemáticos e computacionais.

3. Definindo os fundamentos da Programação Orientada a Objetos

Definir uma estrutura curricular para uma aula não é uma tarefa trivial. O ensino do paradigma POO ainda tem várias dúvidas, associadas a quais conceitos abordar e qual sequência didática é mais adequada. É possível notar que, desde o início da adoção do paradigma em cursos de Computação, a discussão sobre seus fundamentos e métodos de ensino é abordada na literatura. Pugh, LaLonde e Thomas (1987) discutem, em seu trabalho, o impacto da inserção do paradigma POO em cursos de Computação em meados dos 1980, com ênfase no trabalho do educador e como adequar um currículo para os iniciantes.

Uma análise mais estruturada é apresentada por Rosson e Alpert (1990), que definem *Abstração*, *Classe*, *Encapsulamento*, *Ocultamento de Informação*, *Herança*, *Instância*, *Passagem de Mensagem*, *Abstração de dados*, *Método*, *Objeto*, *Modelagem de Objeto* e *Polimorfismo* como os principais conceitos que compõem o paradigma de POO. A análise taxonômica feita por Henderson-Sellers e Edwards (1994) é baseada em um “Triângulo da Orientação a Objetos”, sendo que em cada um dos vértices do triângulo estão os conceitos de *Encapsulamento e Ocultamento de Informação* (*Encapsulation and Information Hiding*), *Abstração* (*Abstraction*) e *Herança e Polimorfismo* (*Inheritance and Polymorphism*).

Armstrong (2006) faz uma análise sobre quais são os fundamentos da POO, através de uma investigação em 239 trabalhos na área pelo termo “desenvolvimento orientado a objetos” (*object-oriented development*). Como resultado, foram definidos 30 conceitos diferentes, e foi verificado que havia conceitos que eram citados em mais de 50% dos estudos, sendo a base de sua taxonomia. Nessa taxonomia, os conceitos foram divididos em “estrutura” (*structure*) e “comportamento” (*behavior*). O grupo “estrutura” é composto por *Abstração*, *Classe*, *Encapsulamento*, *Herança* e *Objeto*, enquanto do grupo “comportamento” é composto por *Método*, *Passagem de Mensagem* e *Polimorfismo*.

Baseando-se nas propostas de Rosson e Alpert (1990), Henderson-Sellers e Edwards (1994) e Armstrong (2006) sobre os fundamentos do paradigma POO, é possível definir uma proposta de currículo que contemple os conceitos fundamentais de POO. Para o método ComFAPOO, foi adotada a taxonomia proposta por Armstrong (2006), que se baseia nas de Rosson e Alpert (1990) e Henderson-Sellers e Edwards (1994), apresentando um embasamento amparado por uma pesquisa sobre uma extensa coleção de trabalhos acadêmicos.

4. O método ComFAPOO

O método ComFAPOO apresenta uma proposta de ensino de POO utilizando as premissas da Aprendizagem Significativa (AS). A AS é uma teoria criada por David Ausubel que se guia pela apresentação gradativa de conceitos para a aquisição do conhecimento por parte do aluno [Ausubel 2003]. Segundo essa teoria, a estrutura cognitiva do indivíduo é formada pela interação de novo conhecimentos com aqueles

que ele já possui ou que pode ser apresentado inicialmente pelo professor (ideia-âncora ou *subsunçor*) [Moreira 2006].

A ComFAPOO também usa os recursos Computação Física (CF) e o design instrucional do *Concreteness Fading* (CoFa). Segundo Desportes (2018), a CF utiliza um conjunto de recursos de hardware que podem ser programáveis, podendo trazer soluções que estimulem o aluno a interagir com o mundo ao seu redor, engajando e estimulando a criatividade e a resolução de problemas por meio de projetos. O *framework* instrucional CoFa segue a premissa da progressão gradual da representação de algo concreto para mais abstrato, fornecendo ao aluno algo mais representativo e tátil antes de apresentar formas mais abstratas [Bruner 1996]. A ComFAPOO é baseada em quatro etapas (*Concepção, Desenvolvimento, Aplicação e Avaliação*), apresentadas com detalhes em Zanetti, Borges e Ricarte (2021). Essas etapas são apresentadas na Figura 1.

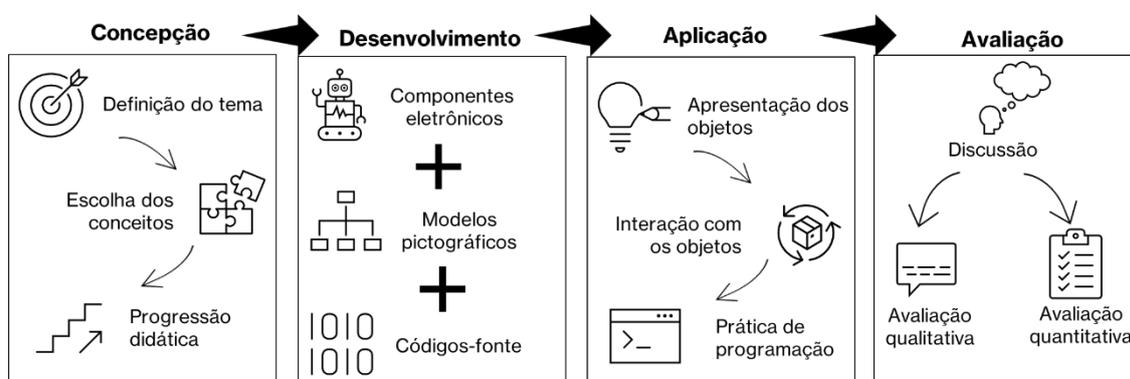


Figura 1. Etapas do método ComFAPOO [Zanetti, Borges e Ricarte 2021].

Segundo Bradley e Lang (1994), o método SAM consiste em um sistema baseado em classificação pictográfica com o objetivo de avaliar diretamente a Satisfação, Motivação e Domínio em resposta a um evento. Essa avaliação foi integrada à ComFAPOO, permitindo que se possa identificar fatores emocionais relacionados à prática didática realizada. A Figura 2 mostra as escalas utilizadas em questionários do método SAM.

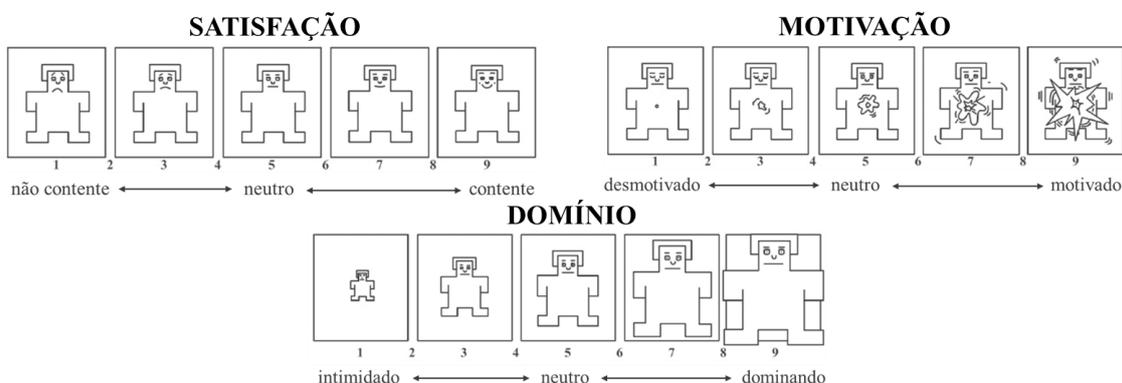


Figura 2. Escalas para os questionários. Adaptado de Soares et al. (2013)

Como instrumento de avaliação, também é proposto um questionário para averiguar a percepção dos alunos com relação aos conceitos fundamentais de POO antes e depois das práticas (pré e pós). Esse questionário é composto por afirmações que

devem ser respondidos por meio da escala Likert, que associa os valores de 1 a 5 às relações de discordância (*discordo totalmente* ou *discordo parcialmente*), neutralidade (*indiferente*) e concordância (*concordo totalmente* ou *concordo parcialmente*) [Severino Júnior e Costa 2014].

Após a aplicação do questionário inicial, as oficinas que compõem a ComFAPOO têm início com a apresentação de objetos concretos. A ComFAPOO sugere a aplicação de oficinas utilizando a plataforma de Arduino, desenvolvida para facilitar o desenvolvimento e programação de sistemas automatizados [McRoberts 2018]. O Arduino é utilizado tanto para facilitar a demonstração das propriedades e ações dos objetos que serão apresentados, como para as práticas de programação.

No início de cada oficina são introduzidos os objetos que serão abstraídos, através da apresentação de suas propriedades e ações como, por exemplo, mostrar que um LED (*Light-Emitting Diode*) muda de estado (aceso e apagado) e pode ter as ações de “acender” e “apagar”. Depois da apresentação do objeto, é discutido quais são as propriedades que esse objeto pode ter ou assumir, fazendo um exercício de abstração, conduzindo a discussão da construção da classe que representa o objeto. Por fim, é exibido o código-fonte que representa esses comportamentos e características e se faz uso dos objetos abstratos, ou seja, os objetos na programação.

Como exemplo, a prática que se baseia na apresentação do objeto LED para demonstrar a relação de classe e objeto, se inicia com a demonstração de quais características que o LED possui, como seu estado (ligado ou desligado), assim como ações que ele pode realizar, como acender, apagar e piscar. Além da demonstração real do objeto e a percepção disso pelo aluno, também é utilizado material gráfico, como sendo um primeiro estágio de transição do objeto concreto para uma representação mais abstrata. Após esse levantamento de características e ações, é feita a representação da classe (representação mais abstratas), através de seus atributos e métodos. Após essa transição, é demonstrado um exemplo de código-fonte que represente a classe, mostrando suas relações com os modelos anteriores. Por fim, é feita a instância do objeto, fazendo com que aquele modelo abstrato controle o objeto real, ilustrando o uso de seus atributos e métodos.

A aplicação de cada uma das partes da ComFAPOO, além de seguir uma sequência gradativa de conceitos conforme recomenda a AS, também de baseou no *framework* CoFa para a apresentação dos objetos concretos até sua representação mais abstrata. O estágio *Ativo* da CoFa auxilia os alunos a interpretar, interagir e compreender as características de um objeto concreto (real); o estágio *Icônico* tem como objetivo retirar características que se desnecessárias ou mal interpretadas, e introduz símbolos formais e faz a ponte entre a representação mais concreta e a mais abstrata; e o estágio *Simbólico*, o mais abstrato, apresenta os conceitos compreendidos usando uma linguagem mais formal e atesta a ação cognitiva da aprendizagem [Bruner 1996]. A Figura 3 apresenta um exemplo da apresentação do objeto LED relacionando as etapas do *framework* CoFa.

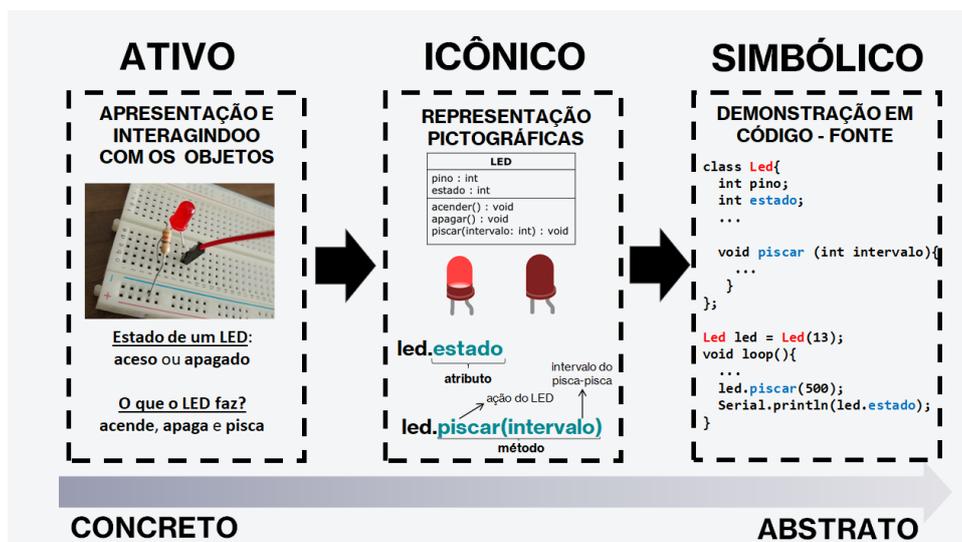


Figura 3. A aplicação do *framework* CoFa. Adaptado de Zanetti e Borges [2021]

A oficina do método ComFAPOO é definida com uma divisão do conteúdo em cinco partes. Cada uma das partes apresenta o conteúdo de forma gradual, correlacionando os conceitos e trazendo significado, seguindo as orientações da AS. Cada uma dessas partes é estruturada para apresentar um ou mais conceitos da POO e, entre elas, também foi planejada uma progressão gradual entre esses conceitos. A escolha dos conceitos para estruturar o currículo da oficina foi baseada nas análises apresentadas na Seção 3 deste artigo. A Tabela 1 descreve de maneira breve os conceitos de cada uma das partes e os objetos que são utilizados.

Tabela 1. Relação de conceitos e objetos usados durante a oficina.

Parte	Conceitos abordados	Objetos utilizados
1	Abstração, classe e objeto	LED e botão
2	Encapsulamento	LED e botão
3	Herança	Sensor de luminosidade e sensor de temperatura
4	Polimorfismo	LED e botão
5	Troca de mensagens entre objetos	Semáforo (LED e sensor de luminosidade)

Em cada uma das partes da oficina são realizadas montagens de circuitos com diferentes componentes eletrônicos, para que os alunos possam ver o resultado da programação em tempo real atuando com o objeto real. Também é estimulada a realização de tarefas que demandam a adição de novos componentes e alterações no código-fonte, para que possam experimentar tanto o trabalho com os objetos concretos como com sua programação.

O questionário aplicado antes da aplicação das oficinas (pré-teste) é formado por 12 questões para serem respondidas por meio da escala Likert, variando de *discordo totalmente* (1) a *concordo totalmente* (5). Essas questões têm como objetivo determinar a percepção de domínio e compreensão dos alunos com relação aos conceitos fundamentais de POO abordados durante a oficina. Esse conjunto de questões será referido neste trabalho como QPC (*Questões sobre Percepção dos Conceitos*). Também foram utilizadas as mesmas questões no questionário a ser respondido após a oficina (pós-teste) para verificar se houve alguma melhoria da percepção dos alunos com relação aos conceitos de POO. As questões são apresentadas na Tabela 2.

Tabela 2. Enunciados das Questões sobre Percepção dos Conceitos (QPC).

Enunciados	
QPC1. De forma geral, você possui domínio dos conceitos fundamentais de POO (Classe, Objetos, Herança, Polimorfismo e outros).	QPC7. Você compreende o conceito de Encapsulamento e sua aplicação.
QPC2. Você compreende o conceito de Abstração aplicado à POO.	QPC8. Você compreende o conceito de Herança e sua aplicação.
QPC3. Você compreende o conceito de Classe.	QPC9. Você compreende o conceito de Polimorfismo e sua aplicação.
QPC4. Você compreende o conceito e a utilização de Objetos.	QPC10. Você compreende a necessidade de utilização de mais de um Objeto em um programa orientado a objetos.
QPC5. Você compreende o conceito de método construtor.	QPC11. Você compreende a troca de mensagens entre os Objetos.
QPC6. Você compreende o conceito de instância.	QPC12. Você se sente capaz de desenvolver um programa orientado a objetos.

O questionário pós-teste também contou com nove questões a serem respondidas usando a escala SAM, organizadas para verificar três diferentes aspectos da ComFAPOO: 1) a dinâmica apresentada durante a oficina; 2) o ato de programar/codificar durante as práticas e; 3) os recursos utilizados durante a oficina. Esse conjunto de questões será referido neste artigo como QPE (*Questões sobre a Percepção Emocional*). A Tabela 3 apresenta as questões relacionadas à escala SAM.

Tabela 3. Enunciados das Questões sobre a Percepção Emocional (QPE).

	Satisfação	Motivação	Domínio
Dinâmica	QPE1. Qual o seu nível de Satisfação com relação à dinâmica utilizada no ComFAPOO?	QPE2. Qual o seu nível de Motivação com relação à dinâmica do ComFAPOO?	QPE3. Qual o seu nível de Domínio com relação à dinâmica do ComFAPOO?
Programar	QPE4. Qual o seu nível de Satisfação em programar (codificar) durante as práticas do ComFAPOO?	QPE5. Qual o seu nível de Motivação em programar (codificar) durante as práticas do ComFAPOO?	QPE6. Qual o seu nível de Domínio em programar (codificar) durante as práticas do ComFAPOO?
Recursos	QPE7. Qual é seu nível de Satisfação com relação aos materiais utilizados (recursos didáticos) durante as práticas do ComFAPOO?	QPE8. Qual é seu nível de Motivação com relação aos materiais utilizados (recursos didáticos) usados durante as práticas do ComFAPOO?	QPE9. Qual é seu nível de Domínio com relação aos materiais utilizados (recursos didáticos) usados durante as práticas do ComFAPOO?

5. A aplicação da oficina e experimentação

Esta seção apresenta uma experimentação. As oficinas propostas pelo método foram realizadas com 14 alunos do 3º módulo do curso Técnico em Desenvolvimento de Sistemas da Escola Técnica Rosa Perrone Scavone, situada na cidade de Itatiba-SP. Todos os alunos estavam no último módulo do curso, com idades variando entre 16 e 17 anos, já tendo cursado disciplinas que utilizam o paradigma de POO.

O material didático da oficina foi disponibilizado em um repositório online hospedado no GitHub¹, rede social para desenvolvedores e de gerenciamento de repositórios. As atividades ocorreram nas dependências da escola, durante o horário de aula. Foram realizadas as montagens dos componentes eletrônicos e a programação no ambiente de desenvolvimento do Arduino. Este projeto foi avaliado e aprovado pelo comitê de ética vinculado à Universidade Estadual de Campinas (UNICAMP)².

A oficina iniciou-se com a aplicação do questionário contendo as QPCs para verificar a percepção do aluno sobre seu nível de domínio com relação aos fundamentos de POO, disponibilizado via a ferramenta online Google *Forms*. Seguindo as diretrizes do método ComFAPOO, foram realizadas todas as partes que compõem as atividades didáticas. Após o término das práticas, foi submetido novamente o mesmo questionário, como os mesmos itens avaliados antes da oficina. Também foi acrescido mais um questionário de avaliação, com as QPES, usando o método SAM, para averiguar as respostas emocionais sobre o método aplicada. A Figura 4 mostra imagens dos alunos durante a dinâmica.

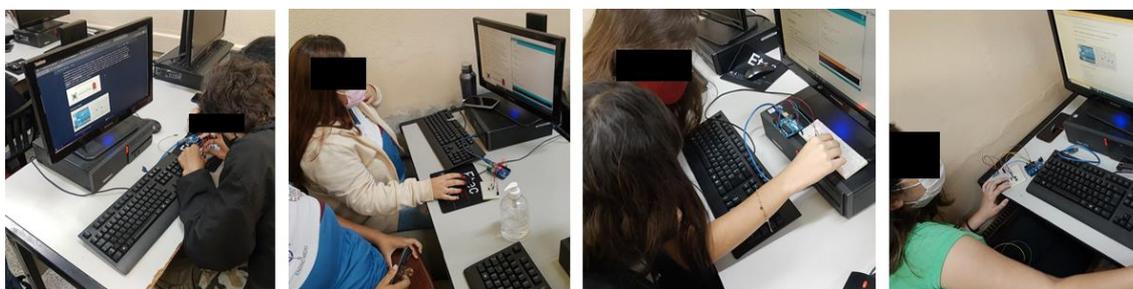


Figura 4. Alunos durante as práticas da oficina.

6. Análise dos resultados

Com relação às QPCs, apresentadas na Tabela 2, a Figura 5 apresenta a dispersão das respostas de cada uma das questões. À esquerda, observa-se o gráfico de barras com a dispersão das respostas (1 a 5) usando a escala Likert no pré-teste. É possível notar que há uma ampla presença de respostas relacionadas a *discordo totalmente*, *discordo parcialmente* e *indiferente*. Isso indica que os alunos apresentavam, antes da aplicação da ComFAPOO, baixa percepção de domínio e compreensão dos conceitos fundamentais de POO. No gráfico de barras à direita, que representa o pós-teste, após a aplicação da ComFAPOO, a maior parte da dispersão de respostas está entre *concordo parcialmente* e *concordo totalmente*.

¹ Repositório disponível em <<https://github.com/humbertozanetti/comfapoo>>

² Certificado de Apresentação de Apreciação Ética (CAAE) de número 29622720.8.0000.5404.

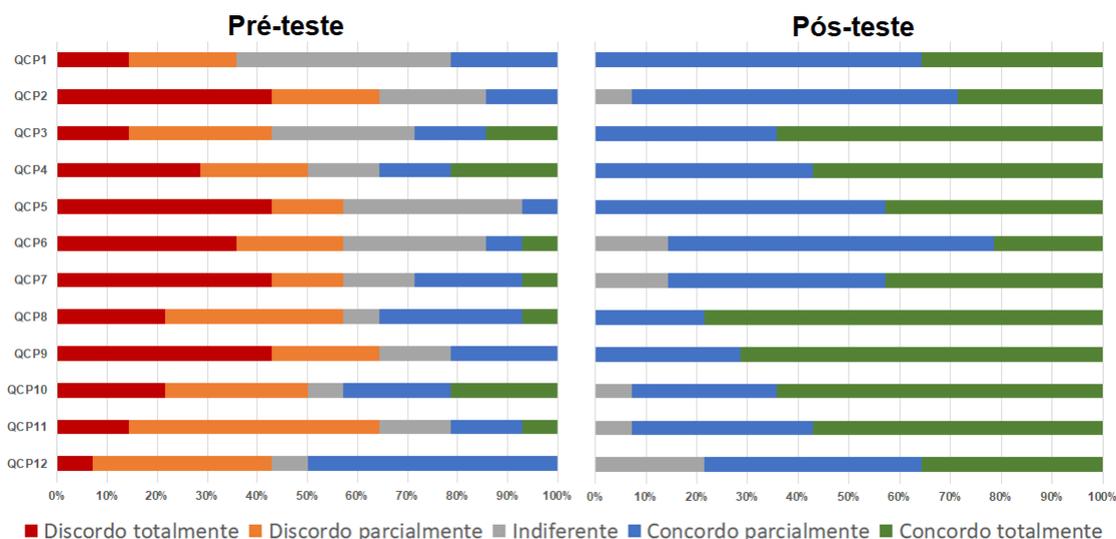


Figura 5. Distribuição das respostas das QPCs.

A partir dos gráficos do pós-teste é possível notar que houve uma melhor percepção por parte dos alunos com relação ao seu conhecimento sobre as bases fundamentais de POO. Houve poucas incidências de *indiferente* e nenhuma de *discordo totalmente* ou *discordo parcialmente*. Com isso, a aplicação da oficina utilizando o método ComFAPOO pode ter proporcionado uma sensível melhoria na compreensão dos fundamentos de POO.

Uma análise mais detalhada mostra que, na questão QPC1, que questionava a compreensão de forma geral sobre o conhecimento do aluno sobre POO, não houve nenhuma avaliação abaixo de *concordo parcialmente*, mostrando que entre todos há uma segurança sobre o que são as bases fundamentais do paradigma. A compreensão dos conceitos de *Classe* e *Objeto*, conceitos que comumente levam a dúvidas logo no início da aprendizagem, também apresenta melhoria na percepção dos alunos após oficina (QPC3 e QPC4). Conceitos tidos como mais complexos, como *Herança* e *Polimorfismo* (QPC8 e QPC9, respectivamente) também tiveram apenas avaliação de concordâncias. Apesar disso, a QPC12, que questiona se o aluno se sente capaz de desenvolver um sistema orientado a objetos, foi a que teve a maior ocorrência de avaliações *indiferente*.

Em relação às QPEs, a Tabela 4 apresenta a distribuição das respostas sobre a escala de nove valores do método SAM. As QPEs estão organizadas pelos diferentes aspectos (*Dinâmica*, *Programar* e *Recursos*) e a incidência de cada uma das escalas (1 a 9). Juntamente com a identificação da QPE, aparece uma indicação se a questão corresponde a (S)atisfação, (M)otivação ou (D)omínio. A última linha da tabela mostra a mediana, para mostrar a tendência central das avaliações por questão.

Com relação ao aspecto referente à *Dinâmica* apresentada na ComFAPOO, é apresentada uma avaliação muito positiva com relação à *Satisfação*, *Motivação* e *Domínio*, com uma predominância com avaliações 7 ou superior. A interação com objetos reais pode ser um grande motivador na execução da prática e no envolvimento do aluno, podendo ter sido o fator mais relevante na composição dessa avaliação

Tabela 4. Respostas às Questões sobre a Percepção Emocional (QPE).

Escala SAM	Dinâmica			Programar			Recursos		
	QPE1 (S)	QPE2 (M)	QPE3 (D)	QPE4 (S)	QPE5 (M)	QPE6 (D)	QPE7 (S)	QPE8 (M)	QPE9 (D)
9	8	4	1	3	4	1	9	8	3
8	3	5	3	3	3	2	3	2	7
7	2	1	6	1	2	5	1	2	3
6	0	2	3	3	3	3	1	1	0
5	1	2	1	3	2	3	0	1	1
4	0	0	0	1	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
Mediana	9,0	8,0	7,0	6,5	7,5	7,0	9,0	9,0	8,0

A percepção sobre o ato de *Programar* foi o que teve a menor avaliação com respeito a *Satisfação*, *Motivação* e *Domínio*. As QPC4, QPC5 e QPC6 apresentaram a maior distribuição de incidência dentre todas as questões. Essa percepção por parte do aluno pode ter sofrido influência da linguagem de programação utilizada. O ato de programar a partir da linguagem do Arduino pode ter influenciado essa percepção do aluno. A plataforma utiliza uma linguagem baseada em C/C++, que possui uma sintaxe mais formal e prolixa quando comparada com a programação em blocos (como Scratch) ou algumas linguagens de programação *script*, como Python.

A percepção sobre os Recursos utilizados teve a avaliação de maior concentração entre os maiores valores da escala. A escolha pelo ferramental da CF para o método ComFAPOO objetivou trazer uma motivação maior ao aluno, por meio de uma percepção mais adequada com a relação entre um objeto concreto e um objeto abstrato, e usando objetos que poderiam ser facilmente compreendidos. Além disso, a facilidade de controlar esses objetos por meio da plataforma Arduino pode ter colaborado com as avaliações positivas nesse quesito. Este resultado pode fornecer subsídios para a adoção de outras plataformas e linguagens, além do Arduino.

7. Considerações finais

O principal objetivo da experimentação relatada neste artigo não foi comprovar a eficiência do método, mas sim ser um estudo de viabilidade, tanto de sua estrutura didática como também na aplicação dos questionários e sua validação por meio da análise desses resultados. De posse dessa primeira avaliação, foi identificada que a aplicação da ComFAPOO é viável e há bons indícios de sua efetividade; o projeto então seguirá fazendo a aplicação da ComFAPOO com um número maior de participantes.

Os resultados preliminares mostram que a ComFAPOO tem potencial para contribuir com a melhoria e compreensão dos fundamentos de POO por parte dos participantes das oficinas, por meio da análise feita sobre as QPCs. Por meio da análise dos QPEs, nota-se uma ampla aceitação com relação à dinâmica e recursos adotados na prática, e potenciais demandas para estudos de novas linguagens e plataformas de programação a serem adotadas.

Como trabalho futuro, é prevista a continuidade da aplicação da oficina com um número maior de participantes e, conseqüentemente, a captação de um maior volume de dados, a fim de desenvolver uma análise estatística mais consistente. Além disso, será analisado a possibilidade de revisão do formato do experimento para que haja grupo de controle com uma abordagem de ensino diferente do método descrito neste trabalho. Também será avaliada a adoção de instrumentos de análise qualitativa e a utilização de novas plataformas de desenvolvimento e linguagens, devido à análise prévia deste trabalho.

Referências

- Abbasi, S., Tabbassum, K., Kazi, H., Tunio, S., & Qureshi, S. (2021). Investigating Student's Obstacles While Learning Object Orientation. *International Journal of Scientific & Technology Research*, 6, 7-11.
- Armstrong, D. J. (2006). The quarks of object-oriented development. *Communications of the ACM*, 49(2), 123-128.
- Ausubel, D. P. (2003). Aquisição e retenção de conhecimentos: uma perspectiva cognitiva. *Lisboa: Plátano, 1*.
- Beck, K., & Cunningham, W. (1989). A laboratory for teaching object oriented thinking. *ACM Sigplan Notices*, 24(10), 1-6.
- Bennedsen, J., & Caspersen, M. E. (2006). Abstraction ability as an indicator of success for learning object-oriented programming?. *ACM Sigcse Bulletin*, 38(2), 39-43.
- Bennedsen, J., & Schulte, C. (2007). What does "objects-first" mean? An international study of teachers' perceptions of objects-first. In *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research-Volume 88* (pp. 21-29).
- Bradley, M. M., & Lang, P. J. (1994). Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, 25(1), 49-59.
- Bruner, J. S. (1966). *Toward a theory of instruction* (Vol. 59). Harvard University Press.
- Desportes, K. S. (2018). *Physical computing education: Designing for student authorship of values-based learning experiences* (Doctoral dissertation, Georgia Institute of Technology).
- Gutiérrez, L. E., Guerrero, C. A., & López-Ospina, H. A. (2022). Ranking of problems and solutions in the teaching and learning of object-oriented programming. *Education and Information Technologies*, 1-35.
- Henderson-Sellers, B., & Edwards, J. (1994). *BOOKTWO of object-oriented knowledge: The working object: Object-Oriented Software Engineering: Methods and Management*. Prentice-Hall, Inc..
- Knudsen, J. L., & Madsen, O. L. (1988). Teaching object-oriented programming is more than teaching object-oriented programming languages. In *European Conference on Object-Oriented Programming* (pp. 21-40). Springer, Berlin, Heidelberg.

- Liberman, N., Beeri, C., & Ben-David Kolikant, Y. (2011). Difficulties in learning inheritance and polymorphism. *ACM Transactions on Computing Education (TOCE)*, 11(1), 1-23.
- Lian, V., Varoy, E., & Giacaman, N. (2022). Learning Object-Oriented Programming Concepts through Visual Analogies. *IEEE Transactions on Learning Technologies*.
- McRoberts, M. (2018). *Arduino básico*. Novatec Editora.
- Moreira, M. A. (2006). Aprendizagem Significativa: da visão clássica à visão crítica (Meaningful learning: from the classical to the critical view). In *Conferência de encerramento do V Encontro Internacional sobre Aprendizagem Significativa, Madrid, Espanha, setembro de sn*.
- Moström, J. E., Boustedt, J., Eckerdal, A., McCartney, R., Sanders, K., Thomas, L., & Zander, C. (2008, September). Concrete examples of abstraction as manifested in students' transformative experiences. In *Proceedings of the Fourth international Workshop on Computing Education Research* (pp. 125-136).
- Pugh, J. R., LaLonde, W. R., & Thomas, D. A. (1987). Introducing object-oriented programming into the computer science curriculum. In *Proceedings of the eighteenth SIGCSE technical symposium on Computer science education* (pp. 98-102).
- Rosson, M. B., & Alpert, S. R. (1990). The cognitive consequences of object-oriented design. *Human-Computer Interaction*, 5(4), 345-379.
- Severino Júnior, D. D. S., & Costa, F. J. (2014). Mensuração e escalas de verificação: uma análise comparativa das escalas de Likert e Phrase Completion. *PMKT–Revista Brasileira de Pesquisas de Marketing, Opinião e Mídia*, 15(1-16), 61.
- Soares, A. P., Pinheiro, A. P., Costa, A., Frade, C. S., Comesaña, M., & Pureza, R. (2013). Affective auditory stimuli: Adaptation of the international affective digitized sounds (IADS-2) for European Portuguese. *Behavior research methods*, 45(4), 1168-1181.
- Trory, A., Howland, K., & Good, J. (2018). Designing for concreteness fading in primary computing. In *Proceedings of the 17th ACM Conference on Interaction Design and Children* (pp. 278-288).
- Xinogalos, S. (2015). Object-oriented design and programming: an investigation of novices' conceptions on objects and classes. *ACM Transactions on Computing Education (TOCE)*, 15(3), 1-21.
- Zanetti, H. A., & Borges, M. A. (2021). Por que estimular a Aprendizagem Significativa no ensino de Programação Orientada a Objetos?. In *Anais do Simpósio Brasileiro de Educação em Computação* (pp. 290-295). SBC.
- Zanetti, H. A. P., Borges, M. A. F. & Ricarte, I. L. M. (2021). Método de ensino de programação Orientada a Objetos baseado em Computação Física, Aprendizagem Significativa e Concreteness Fading. In A. J. Osório, M. J. Gomes, A. Ramos, & A. L. Valente (Eds.), *Challenges 2021, desafios do digital: Livro de atas* (pp. 23-32). Universidade do Minho. Centro de Competência.