ATIVIDADE DO GRUPO DE EDUCAÇÃO TUTORIAL NA DISCIPLINA DE INTRODUÇÃO À ENGENHARIA COMPUTACIONAL

DOI: 10.37702/2175-957X.COBENGE.2024.5121

Autores: *JOVENTINO DE OLIVEIRA CAMPOS, MARCELO IAN REZENDE MENEZES, LUIZ GUSTAVO FERREIRA NAZARETH, JOAO VICTOR COSTA DE OLIVEIRA, AUGUSTO DE FILIPPO CAVALINI, JOAO STEPHAN SILVA MAURICIO, GABRIEL MATTOS, DAVI LUIS DE FARIA ROCHA, JULIA ZOFFOLI CACADOR, ROBERT GONCALVES VIEIRA DE SOUZA, HIERO HENRIQUE BARCELOS COSTA, GABRIEL HENRIQUE DE SOUZA*

Resumo: *O Grupo de Educação Tutorial da Engenharia Computacional (GET-EngComp) tem como objetivo acompanhar, orientar e auxiliar alunos do curso de Engenharia Computacional, bem como desenvolver as capacidades individuais e coletivas de seus membros. Nesse sentido, em parceria com a coordenação do curso, o GET-EngComp é responsável por ministrar uma pequena parcela das aulas da disciplina de Introdução à Engenharia Computacional. São elaborados exercícios e aulas de conteúdos relevantes à formação acadêmica na Engenharia. O objetivo da atividade envolve buscar o interesse dos alunos, geralmente calouros, ao curso, mostrando aplicações práticas dos conteúdos vistos em aula, além de introduzir o funcionamento do GET-EngComp a eles, buscando maior reconhecimento ao grupo. As atividades desenvolvidas com os alunos envolveram o ensino da linguagem de programação Python integrado ao aprendizado de conceitos de modelagem computacional, métodos numéricos e análise de dados. O presente trabalho apresenta um relato da experiência do grupo com esta atividade para três turmas diferentes da disciplina, apresentando o desempenho dos alunos, o aprimoramento da atividade e as lições aprendidas.*

Palavras-chave: *modelagem computacional, métodos numéricos, programação em python*

ATIVIDADE DO GRUPO DE EDUCAÇÃO TUTORIAL NA DISCIPLINA DE INTRODUÇÃO À ENGENHARIA COMPUTACIONAL

1 INTRODUÇÃO

O curso de Engenharia Computacional da Universidade Federal de Juiz de Fora (UFJF) é um curso interdisciplinar voltado para a área de modelagem computacional. O principal objetivo do currículo do curso é preparar os alunos para serem capazes de formular, analisar, implementar e aplicar modelos matemáticos, métodos numéricos e sistemas computacionais para resolver problemas científicos e de engenharia (LOBOSCO *et al.* 2019). O curso possui um grupo de educação tutorial (GET) fomentado pela UFJF que segue os moldes do Programa de Educação Tutorial (PET) financiado pelo Ministério da Educação. Neste grupo, alunos sob a tutoria de um professor realizam atividades de ensino, pesquisa e extensão, buscando uma melhor formação para os alunos do curso, conforme (MARTINS, 2007) e (MOTA *et al.*, 2019).

Diversos cursos da graduação possuem uma disciplina para introduzir os alunos aos diversos conceitos e atividades que a universidade oferece, bem como ajudá-los a conhecer o curso no qual estão inseridos. Nesse contexto, para estudantes de Engenharia Computacional, há a obrigatoriedade de cursar a disciplina de Introdução à Engenharia Computacional a partir do segundo período do curso.

Dessa forma, a convite dos coordenadores do curso, o Grupo de Educação Tutorial da Engenharia Computacional (GET-EngComp) realiza atividades na disciplina, incluindo apresentações das atividades do grupo, da grade do curso e a realização de uma atividade de capacitação com os alunos. Na maioria das vezes, esta atividade envolve o ensino da linguagem de programação *Python* no contexto de computação científica, de forma que os alunos possam realizar atividades relacionadas a métodos numéricos, assunto pertinente ao curso em questão. Diferentemente de uma monitoria ou plantão de dúvidas, onde o foco é revisar o conteúdo dado em sala de aula pelo professor, o GET-EngComp propõe e ministra aulas sobre os conteúdos mencionados acima, buscando trazer um material diferente daquele passado pelo professor.

O projeto tem como principais objetivos apresentar as atividades do grupo aos alunos matriculados, incentivando-os a participar de atividades extracurriculares para enriquecer sua experiência acadêmica. Já a atividade com a turma, visa introduzir os principais conceitos e ferramentas do curso, como modelagem computacional, programação em Python e visualização de dados, para melhor compreensão da proposta do curso e capacidade de resolver problemas computacionalmente. No final, espera-se que os alunos fiquem interessados na graduação de forma a diminuir a evasão e motivá-los também a fazer parte do GET-EngComp.

Portanto, este trabalho apresenta as atividades do GET-EngComp na disciplina de de Introdução à Engenharia Computacional nos últimos anos, avaliando o envolvimento dos alunos com as atividades e com o grupo. O restante do artigo está assim estruturado: a seção 2 apresenta a fundamentação teórica, a seção 3 possui a metodologia empregada no projeto, a seção 4 apresenta o relato de experiência e os resultados, na seção 5 se encontra a análise dos estudos e na seção 6 apresenta-se as considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

A Engenharia muitas vezes é ensinada de forma dedutiva, com conteúdos teóricos que progridem para as suas aplicações. Muitos alunos acabam ficando desmotivados nos estudos por não entenderam a importância dos princípios aprendidos em disciplinas básicas para a formação acadêmica. Uma alternativa citada por PRINCE E FELDER (2006) é a do aprendizado indutivo, onde é apresentado aos alunos uma análise de caso, um problema da vida real ou um conjunto de dados para serem interpretados. Tal método se mostra eficaz no ensino por incentivar os alunos a discutirem sobre os problemas apresentados e colaborarem entre si.

A modelagem computacional, presente na Engenharia Computacional, é uma disciplina fundamental que permite aos engenheiros representar sistemas físicos e processos de engenharia por meio de modelos matemáticos e simulações computacionais. Ao revisar estudos como o de LOBOSCO *et al*. (2018), observa-se que a modelagem computacional não apenas fornece uma visão abrangente dos métodos e técnicas utilizados para resolver problemas complexos em diversas áreas da engenharia, mas também ajuda os alunos a desenvolverem habilidades analíticas e de resolução de problemas essenciais para sua formação como engenheiros computacionais. Assim, encoraja-se uma participação ativa dos alunos na resolução de problemas, não se limitando ao aprendizado teórico.

Em um estudo conduzido por SANTOS E MENDES (2013) na Universidade de Coimbra implementou-se um programa de introdução à programação para estudantes de Engenharia, com foco no ensino de linguagens de programação e suas aplicações em problemas reais de engenharia. Os resultados mostraram que uma abordagem prática e contextualizada contribui significativamente para o engajamento dos alunos e para a compreensão dos conceitos fundamentais de programação e modelagem computacional.

Por sua vez, os métodos numéricos desempenham um papel crucial na resolução de problemas matemáticos e de engenharia que não possuem soluções analíticas exatas. Estudos como o de MCKINNEY (2017) ressaltam a importância de compreender os princípios subjacentes e as limitações computacionais associadas a esses métodos. Ao introduzir os alunos a métodos numéricos durante as aulas de Introdução à Engenharia Computacional, os estudantes são capacitados a aplicar essas técnicas na resolução de problemas do mundo real, preparando-os para enfrentar desafios complexos durante sua carreira profissional.

1 METODOLOGIA

A participação do GET-EngComp na disciplina de Introdução à Engenharia Computacional desempenha um papel crucial na formação dos alunos, promovendo uma compreensão profunda dos princípios fundamentais da modelagem computacional aplicada à engenharia. Nesse tópico é abordada a metodologia empregada na atividade, incluindo o material utilizado, a aplicação e a avaliação.

3.1 Conceitos básicos

A disciplina se encontra no segundo período da matriz curricular do curso, momento em que os alunos já tiveram contato com conteúdos de cálculo e programação. Dessa forma, as atividades desenvolvidas com os alunos envolvem o ensino da linguagem de programação *Python* integrado ao aprendizado de conceitos de modelagem computacional, métodos numéricos e análise de dados.

Python é uma linguagem de programação de alto nível amplamente utilizada em ciência e engenharia, devido à sua sintaxe clara, flexibilidade e vasta coleção de bibliotecas especializadas. A compreensão dos princípios básicos de programação em *Python* é fundamental para implementar soluções computacionais eficazes em diversas áreas da engenharia (VAN ROSSUM *et al.* 2009).

A modelagem computacional envolve a representação matemática de sistemas físicos e processos de engenharia por meio de modelos matemáticos, algoritmos e simulações computacionais. Essa abordagem permite aos engenheiros analisar o comportamento de sistemas complexos, prever seu desempenho e otimizar seu projeto antes da implementação prática (SHIFLET *et al.* 2014).

Os métodos numéricos são técnicas utilizadas para resolver problemas matemáticos complexos que não possuem soluções analíticas exatas. A aplicação eficaz desses métodos requer uma compreensão profunda dos princípios matemáticos subjacentes e das limitações computacionais associadas (FRANCO, 2006).

A análise de dados é uma parte essencial da Engenharia Computacional, permitindo aos engenheiros extrair conhecimentos úteis a partir de conjuntos de dados complexos. A biblioteca Pandas oferece estruturas de dados poderosas e ferramentas para manipulação, limpeza e análise de dados, facilitando a exploração e interpretação de informações em contextos científicos e tecnológicos (MOLIN, 2021).

3.2 Tópicos abordados

Até o segundo semestre de 2022, o GET estruturou suas atividades em torno de dois eixos principais: introdução à programação utilizando a linguagem Python e suas bibliotecas, e uma abordagem prática na resolução numérica de equações diferenciais, empregando o método de Euler.

De forma a aumentar o entendimento dos alunos na implementação de métodos numéricos, a partir do segundo semestre de 2023 optou-se por substituir a atividade de solução numérica de equações diferenciais por uma focada na solução numérica de integrais. Essa mudança estratégica não apenas amplia o escopo de aplicação dos métodos numéricos, mas também estabelece uma conexão mais direta entre os conceitos aprendidos em cursos de cálculo e física e suas implementações práticas por meio de ferramentas computacionais.

Além disso, o GET expandiu suas atividades para incluir uma componente dedicada ao estudo da linguagem de programação *Python* e uma de suas bibliotecas, chamada *Pandas*. Em colaboração com o corpo docente da disciplina, o grupo ministra uma aula especializada sobre o uso dessa biblioteca, destacando sua ampla aplicabilidade na organização, filtragem e análise de dados qualitativos e quantitativos.

No segundo semestre de 2022, essa atividade envolveu a análise detalhada do crescimento dos preços de alugueis de imóveis nos Estados Unidos, assim como a investigação do crescimento da população HIV positivo em cada país membro da Organização Mundial da Saúde (OMS). Já no segundo semestre de 2023, a abordagem foi ajustada para incluir uma análise qualitativa do trabalho de diretores de cinema, explorando os filmes mais bem avaliados e lucrativos de cada diretor analisado. Essas análises, além de demonstrarem a versatilidade da biblioteca *Pandas*, proporcionaram aos alunos uma compreensão mais abrangente da aplicação prática de técnicas computacionais em diferentes domínios.

3.3 Material

O material apresentado na atividade é desenvolvido e discutido pelo GET antes de ser ministrado aos alunos da disciplina. A atividade leva em consideração que a sala de

aula é equipada com computadores, portanto possuem caráter prático. As aulas possuem uma exposição inicial usando slides e posteriormente a parte prática é desenvolvida usando a ferramenta Jupyter Notebook, através do ambiente Google Colaboratory (TOCK, 2019). Esta ferramenta foi escolhida por ser armazenada em nuvem, necessitando apenas de um computador e conexão à internet para ser utilizada, sendo intuitivo aos alunos que estão aprendendo sobre programação.

No conteúdo apresentado inicialmente tem-se um minicurso de *Python* básico para apresentar aos alunos a sintaxe da linguagem e algumas bibliotecas úteis, criando uma base para que os alunos possam se familiarizar com o conteúdo. Posteriormente, uma introdução a algum método numérico mais básico é realizada, para que em outra aula seja possível trabalhar no desafio proposto, que seria relativamente mais complexo que os exemplos. Também são apresentados exercícios para se resolver em sala de aula, para que os alunos possam tirar eventuais dúvidas. Muitos destes exercícios são adaptados do livro-texto apresentado na disciplina, e outros, possivelmente mais complexos, estão relacionados à Equações Diferenciais Ordinárias, com base no que é apresentado em ZILL (2016). Este conteúdo é dividido entre 3 e 5 aulas de 2h, a depender do semestre letivo.

3.4 Avaliação

Para avaliar os alunos, foram desenvolvidas atividades avaliativas que englobam o conteúdo apresentado. Com o prazo de ao menos uma semana, os alunos são desafiados a desenvolver soluções para um conjunto de problemas elaborados pelo próprio GET usando programação em *Python*. Uma das atividades envolve a implementação de um método numérico para a solução de problemas e outra atividade consiste em análises realizadas em uma base de dados disponibilizada, buscando extrair certas informações.

Ao fim da disciplina, o GET também enviou um formulário de avaliação para conferir a opinião dos alunos matriculados. São cinco perguntas onde os alunos devem responder com base na escala Likert, de 1 a 5:

- Como você avalia a atividade?
- O material apresentado foi útil?
- A matéria foi explicada de forma clara?
- Você compreendeu a atividade proposta?
- Você conseguiu fazer a atividade proposta?

No final do formulário também há um espaço para sugestões e opiniões gerais. Esses dados foram coletados e analisados na próxima seção.

4 RELATO DE EXPERIÊNCIA

O presente estudo investigou os resultados obtidos a partir da implementação de diferentes metodologias de ensino na disciplina de Introdução à Engenharia Computacional, com o objetivo de analisar o impacto dessas estratégias no desempenho acadêmico dos alunos. A análise focou especificamente nas atividades já mencionadas anteriormente. Por meio de uma abordagem quantitativa, buscou-se compreender como essas intervenções influenciaram a compreensão e o domínio dos conceitos fundamentais da engenharia computacional pelos estudantes.

Inicialmente, os resultados obtidos nas atividades de solução numérica de equações diferenciais revelaram uma média da nota da turma em torno de 68%. Entretanto, após a transição para uma abordagem que priorizava a solução numérica de integrais, observou-se um aumento significativo na média da turma, alcançando o valor de

85%. Esse incremento substancial pode ser atribuído, em grande parte, à capacidade dos alunos de estabelecer conexões entre os conceitos teóricos aprendidos em disciplinas como cálculo e física e sua aplicação prática por meio de métodos numéricos, evidenciando uma compreensão mais profunda e abrangente dos princípios envolvidos.

No que diz respeito às atividades de programação em *Python*, os resultados mostraram uma evolução consistente ao longo dos semestres. Inicialmente, no segundo semestre de 2021, a média da turma foi de 71%, registrando um aumento para 75% no semestre seguinte (2022) e atingindo o valor notável de 88% no semestre subsequente (2023). Essa progressão ascendente na pontuação reflete o desenvolvimento gradual das habilidades dos alunos na manipulação de algoritmos e estruturas de dados em *Python*, culminando em uma maior proficiência na resolução de problemas computacionais complexos.

Por outro lado, as atividades relacionadas ao uso da biblioteca *Pandas* apresentaram uma média de pontuação próxima a 73,5%. Embora tenha havido uma melhora em relação aos semestres anteriores, esse resultado indica que ainda há espaço para aprimoramento no entendimento e aplicação dessa ferramenta. Isso ressalta a importância de uma abordagem mais abrangente e direcionada ao estudo da *Pandas*, visando explorar todo o seu potencial na organização, filtragem e análise de dados quantitativos e qualitativos em contextos científicos e tecnológicos.

Assim, os resultados desta investigação destacam a eficácia das estratégias de ensino adotadas na disciplina de Introdução à Engenharia Computacional. O aumento na média da turma nas atividades de solução numérica de integrais e programação em Python reflete o impacto positivo de uma abordagem prática e contextualizada na aprendizagem dos alunos. Contudo, a constatação de que ainda há margem para melhorias no entendimento e aplicação da biblioteca *Pandas* destaca a importância contínua do desenvolvimento e aprimoramento das metodologias de ensino empregadas no campo da engenharia computacional.

O formulário enviado recentemente aos alunos que cursaram a disciplina nos três últimos anos contou com a resposta de 11 estudantes. Dos alunos matriculados em Introdução à Engenharia Computacional em 2021, onde a atividade abordou a solução numérica de equações diferenciais, a média da avaliação da atividade foi de 4.4 (escala de 1 a 5), evidenciando que os alunos gostaram da proposta. No entanto, no quesito sobre entendimento da atividade, a média caiu para 4. No ano seguinte (2022) houve uma queda significativa na avaliação dos alunos, chegando ao valor de 2.5. Portanto, os alunos não consideraram a atividade interessante, apesar da nota geral ter aumentado. Inclusive, um dos alunos comentou que a atividade estava confusa para uma matéria dada no segundo período. Finalmente, no ano de 2023, a média subiu novamente em relação aos outros dois anos, chegando a 5 pontos, mostrando que a mudança de foco para métodos numéricos aplicado ao cálculo de integrais e bibliotecas em *Python* aumentou o interesse dos alunos, ao mesmo tempo que promoveu o contato entre eles.

5 LIÇÕES APRENDIDAS

As atividades propostas pelo GET na disciplina de Introdução à Engenharia Computacional promovem o aprendizado não só dos alunos matriculados, como também dos membros do próprio grupo. Essas lições abrangem desde aspectos relacionados à metodologia de ensino até a interação entre os membros do grupo e os estudantes. Nesta seção são destacadas algumas das principais lições aprendidas.

5.1 Flexibilidade na Abordagem Pedagógica

Uma das lições mais importantes foi a necessidade de adaptar constantemente a abordagem didática para atender aos requisitos dos alunos e às demandas do curso. A transição da atividade de solução numérica de equações diferenciais para a solução numérica de integrais exemplifica essa flexibilidade, evidenciando a importância de avaliar e ajustar as estratégias de ensino com base no *feedback* dos alunos e nas tendências educacionais.

Ao perceber que uma mudança no foco das atividades poderia melhorar a compreensão dos alunos e tornar o conteúdo mais relevante, o GET-EngComp demonstrou sua capacidade de adaptar-se rapidamente às necessidades em evolução do grupo. Isso enfatiza a importância de avaliar continuamente as estratégias de ensino com base no feedback dos alunos e nas tendências educacionais, garantindo que o material apresentado seja relevante e eficaz.

5.2 Importância da Interdisciplinaridade

A introdução de uma atividade dedicada ao estudo da biblioteca Pandas ressaltou a importância da interdisciplinaridade no ensino da Engenharia Computacional. O uso dessa biblioteca não apenas ampliou o escopo de aplicação dos métodos numéricos, mas também permitiu aos alunos explorar o potencial da programação em *Python* na análise de dados quantitativos e qualitativos em diversos contextos científicos e tecnológicos.

Isso não apenas ampliou o escopo de aplicação dos métodos numéricos, mas também demonstrou como a interdisciplinaridade pode enriquecer a experiência educacional, permitindo aos alunos aplicar conceitos de programação em um contexto real e relevante.

5.3 Feedback como Ferramenta de Aprimoramento

A coleta e análise do *feedback* dos alunos por meio de formulários de avaliação proporcionaram a compreensão sobre os pontos fortes e fracos das atividades desenvolvidas pelo GET-EngComp. Esse processo demonstrou a importância de estabelecer uma comunicação aberta e transparente com os alunos, incentivando-os a expressar suas opiniões e sugestões para o aprimoramento contínuo das práticas de ensino, permitindo assim, a revisão da nossa didática.

Entre uma das maiores dificuldades encontradas pelo GET-EngComp destaca-se a necessidade de adaptar materiais e a metodologia de ensino para nivelar a passagem de conteúdo das aulas, garantindo que estudantes de diferentes níveis de conhecimento possam acompanhar o desenvolvimento de raciocínios mais facilmente. Fatores como a complexidade dos temas abordados e a falta de engajamento dos alunos foram temas que foram levantados ao longo do decorrer das discussões e atividades.

5.4 Desenvolvimento de Habilidades de Comunicação

A interação regular com os alunos durante as atividades do GET-EngComp ajudou os membros do grupo a desenvolver habilidades de comunicação eficazes. Essa experiência não apenas fortaleceu a capacidade dos membros do grupo de transmitir conceitos complexos de forma clara e acessível, mas também incentivou a colaboração e o trabalho em equipe entre os membros do GET.

Ao enfrentar desafios e resolver dúvidas dos alunos em conjunto, os membros do grupo aprenderam a trabalhar de forma colaborativa, compartilhando conhecimentos e experiências para alcançar objetivos comuns. Essa colaboração enriqueceu a experiência de aprendizado, permitindo que os alunos se beneficiassem da variedade de perspectivas

e habilidades presentes no GET-EngComp.

6 Considerações FINAIS

A partir da análise dos resultados e das lições aprendidas, é possível notar que a abordagem prática e contextualizada adotada pelo GET-EngComp teve um impacto positivo no desempenho acadêmico dos alunos e na sua compreensão dos conceitos fundamentais da Engenharia Computacional. A flexibilidade na metodologia de ensino, aliada à interdisciplinaridade e à colaboração com o corpo docente, permitiu adaptar as atividades às necessidades específicas dos alunos, promovendo um ambiente de aprendizado dinâmico e estimulante.

No entanto, há espaço para aprimoramentos. O *feedback* dos alunos destacou áreas que podem ser desenvolvidas, como a ampliação do escopo de aplicação da biblioteca Pandas e a diversificação das atividades práticas. Além disso, a contínua busca por estratégias inovadoras e recursos didáticos atualizados é essencial para acompanhar as demandas em constante evolução da área de Engenharia Computacional.

AGRADECIMENTOS

Agradecemos à Universidade Federal de Juiz de Fora, pelo financiamento dos bolsistas do GET-EngComp e pela oferta do curso de Engenharia Computacional. Além disso, agradecemos aos professores da disciplina e coordenadores do curso, Ruy Freitas Reis e Elson Magalhães Toledo, pelo espaço concedido ao GET-EngComp para o realização destas atividades.

REFERÊNCIAS

FRANCO, Neide Maria Bertoldi Franco. **Cálculo Numérico**. [S. l.]: Pearson, 2006. 520 p.

LOBOSCO, Marcelo et al. **Design and Analysis of an Undergraduate Computational Engineering Degree at Federal University of Juiz de Fora**. In: International Conference on Computational Science. Cham: Springer International Publishing, 2018. p. 293-303.

MARTINS, Iguatemy Lucena. Educação Tutorial no ensino presencial: uma análise sobre o PET. **PET–Programa de Educação Tutorial: estratégia para o desenvolvimento da graduação. Brasília: Ministério da Educação**, 2007.

MCKINNEY, William. **Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython.** 2. ed, 2017.

MOLIN, Stefanie. **Hands-On Data Analysis with Pandas**: a python data science handbook for data collection, wrangling, analysis, and visualization. 2. ed. [S. l.]: Packt Publishing, 2021. 788 p.

MOTA, André Felipe de Souza *et al*. **Sobre a atuação na graduação dos GETianos da Engenharia Computacional**. Rio de Janeiro: Sudeste Pet, 2019.

PRINCE, Michael & FELDER, Richard. (2006). Inductive Teaching and Learning Methods: Definitions, Comparisons, and Research Bases. Journal of Engineering Education. 95. 123-137. 10.1002/j.2168-9830.2006.tb00884.x.

SANTOS, Gomes, A., e MENDES, A. (2013). "**A taxonomy of exercises to support individual learning paths in initial programming learning**." Proceedings of the XYZ Conference on Engineering Education.

SHIFLET, Angela B.; SHIFLET, George W.. **Introduction to Computational Science**: modeling and simulation for the sciences. 2. ed. Princeton, NJ: Princeton University Press, 2014. 856 p.

TOCK, Kalée. **Google CoLaboratory as a platform for Python coding with students.** RTSRE Proceedings, v. 2, n. 1, 2019.

VAN ROSSUM, Guido *et al*. **Python 3 Reference Manual**. Scotts Valley, CA: Createspace, 2009. 242 p.

ZILL, Dennis G.. **Equações Diferenciais**: com aplicações em modelagem. 3. ed. [S. I.]: Editora Cengage, 2016. 504 p. Tradução da 10ª edição norte-americana.

ACTIVITY OF THE TUTORIAL EDUCATION GROUP IN THE INTRODUCTION TO COMPUTATIONAL ENGINEERING DISCIPLINE

Abstract: *The Computational Engineering Tutorial Education Group (GET-EngComp) aims to monitor, guide and assist students in the Computer Engineering course, as well as develop the individual and collective capabilities of its members. Therefore, in partnership with the course coordination, GET-EngComp is responsible for teaching a small portion of the classes in the Introduction to Computational Engineering discipline. In this activity there are elaborate exercises and classes with content relevant to academic training in Engineering. The goal of the activity involves seeking the interest of students, generally freshmen, showing practical applications of the content seen in class, in addition to introducing the operation of GET-EngComp to them, seeking greater visibility for the group. The activities developed with the students involved teaching the Python programming language integrated with learning concepts of computational modeling, numerical methods and data analysis. This work presents a report on the group's experience with this activity for three different classes of the subject, presenting the students' performance, the improvement over the years and the lessons learned.*

Keywords: *computational modeling, numerical methods, python programming*