



**MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE MINAS GERAIS  
PRÓ-REITORIA DE EXTENSÃO**

**PROJETO PEDAGÓGICO DO CURSO DE FORMAÇÃO CONTINUADA  
Programador Python: Nível Intermediário<sup>1</sup>**

**Modalidade EaD**

**Piumhi - MG**

**Junho/2020**

---

<sup>1</sup> Nomenclatura baseada no Código Brasileiro de Ocupações CBO 3171-10 (“Programador de Sistemas de Computador”) e associada ao Catálogo Nacional de Cursos Técnicos (“Técnico em Desenvolvimento de Sistemas”)



**MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE MINAS GERAIS  
PRÓ-REITORIA DE EXTENSÃO**

|                                |                              |
|--------------------------------|------------------------------|
| <b>Reitor:</b>                 | Kléber Gonçalves Glória      |
| <b>Pró-Reitor de Extensão:</b> | Carlos Bernardes Rosa Júnior |
| <b>Diretor do campus:</b>      | Humberto Coelho de Melo      |
| <b>Coordenador do curso:</b>   | Stella Maria Gomes Tomé      |

**PROJETO PEDAGÓGICO DO CURSO DE FORMAÇÃO CONTINUADA  
Programador Python: Nível Intermediário**

**Modalidade EaD**

Projeto Pedagógico do curso “Programador Python: Nível Intermediário”, submetido ao Setor de Extensão do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais, como requisito parcial para a aprovação de Curso de Formação Continuada.

**Piumhi  
Junho/2020**

## Sumário

|   |   |
|---|---|
| 1. Dados institucionais   | 4 |
| 2. Dados gerais do curso  | 4 |
| 3. Justificativa  | 5 |
| 4. Objetivos do curso   | 5 |
| 5. Público-alvo   | 6 |
| 6. Pré-requisitos e mecanismos de acesso ao curso   | 6 |
| 7. Matriz curricular  | 6 |
| 8. Procedimentos didático-metodológicos   | 6 |
| 9. Descrição dos principais instrumentos de avaliação   | 6 |
| 10. Definição dos mínimos de frequência e/ou aproveitamento da aprendizagem para fins de aprovação/certificação | 7 |
| 11. Infraestrutura física e equipamentos  | 7 |
| 12. Referências   | 7 |
| Anexo I – Plano de Ensino   | 8 |

## 1. Dados Institucionais

|                       |  |
|-----------------------|--|
| Razão Social          | Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – IFMG |
| CNPJ                  | 10.626.896/0001-72   |
| Esfera Administrativa | Federal  |
| Endereço              | Rua Severo Veloso, 1880 – Bairro Bela Vista - Piumhi/MG – 37.925-000       |
| Telefone/Fax          | (37)3371-3371-3755   |
| Site da instituição   | www.ifmg.edu.br/piumhi   |

## 2. Dados Gerais do Curso

|                           |  |
|---------------------------|--|
| Nome do curso             | Programador Python: Nível Intermediário  |
| Número de vagas por turma | A definir  |
| Periodicidade das aulas   | Semanal  |
| Carga horária             | 30 h   |
| Modalidade da oferta      | À distância  |
| Local das aulas           | Ambiente Virtual de Aprendizagem   |
| Coordenador do curso      | Stella Maria Gomes Tomé<br>stella.tome@ifmg.edu.br<br><br>Tecnóloga em Informática graduada desde 2003 pela Universidade do Estado de Minas Gerais, Especialista em Informática para Educação pela Universidade Federal de Lavras, Mestre em Economia pela Universidade Federal de Viçosa. |

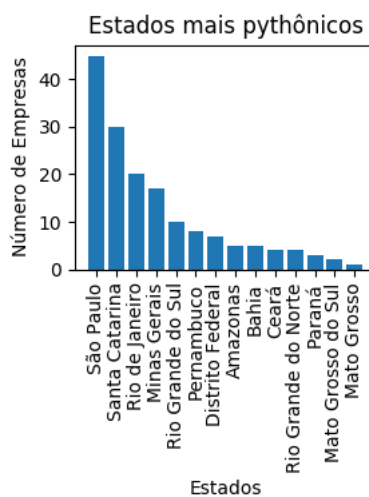
### 3. Justificativa

A linguagem *Python* de programação vem sendo considerada uma das principais linguagens de programação da atualidade. Essa linguagem pode ser usada para desenvolver sistemas: *desktop*, *web* e *scripts*, o que a torna bastante flexível. Muitos *sites* como *Google* e *Globo.com* adotam a linguagem *Python*.

*Python* é um *software* totalmente livre, multiplataforma, ou seja, pode ser usado gratuitamente e seus programas podem ser executados em qualquer sistema operacional que possui suporte, inclusive em celulares.

De acordo com o site oficial do *Python* (2020), cada vez mais os estados brasileiros tem aderido à essa linguagem de programação, otimizando seus trabalhos e criando sistemas cada vez mais inteligentes.

**Figura 1** – Ranking da utilização de Python nos estados brasileiros



Fonte – <https://python.org.br/>, (2020)

A facilidade da sintaxe na utilização do Python permite um maior foco na resolução dos problemas em lugar de se preocupar unicamente com a codificação do sistema. Dessa forma, pode-se salientar o impacto da programação para o desenvolvimento das capacidades cognitivas, indicando conexões entre a programação e solução de problemas lógicos, além de aprimorar o pensamento lógico-matemático.

Portanto, qualquer indivíduo, mesmo que não seguindo a área técnica de informática poderá se beneficiar dessas habilidades aprendidas.

## 4. Objetivos do curso

### 4.1 Objetivo Geral

Proporcionar conhecimentos à programação de computadores na linguagem *Python*.

### 4.2 Objetivos Específicos

1. Capacitar o estudante para desenvolver programas básicos utilizando a linguagem de Programação *Python*, tendo condições e entendimento para continuar o desenvolvimento da linguagem.
2. Contribuir para o processo de formação dos atores inscitos, possibilitando o desenvolvimento de programas em Python.

3. Aplicar a linguagem de programação Python para resolução de problemas científicos e tecnológicos.
4. Possibilitar a compreensão do o paradigma da programação orientada a objeto em Python utilizando a sintaxe da linguagem e características de organização de classes em pacotes e projetos na solução de problemas, assim como a implementação de softwares com recursos gráficos (GUI – *Graphical User Interface*).

## 5. Público-alvo

O curso Programador Python: Nível Intermediário, tem como público alvo principal, indivíduos que já tenham concluído o ensino fundamental tenham intenção em desenvolver habilidades de programação orientada a objetos. Ainda assim, o curso está aberto para pessoas que queiram conhecer a linguagem e se valer de seu uso/desenvolvimento.

## 6. Pré-requisitos e mecanismos de acesso ao curso

Pré-requisitos para seleção: Ensino Fundamental Completo.

Processo seletivo: ordem de inscrição e entrevistas assíncronas.

## 7. Matriz curricular

| Componentes Curriculares |   |
|--------------------------|---|
| 1ª Semana – 10 horas     | i) Iteradores;<br>ii) Recursividade, ordenação e busca;<br>iii) Funções Anônimas;<br>iv) Exemplos Práticos; |
| 2ª Semana – 10 horas     | v) <i>List comprehensions, mapping e filter</i> ;<br>vi) Erros e Exceções;<br>vii) Práticas;                |
| 3ª Semana – 10 horas     | viii) Manipulação de Arquivos;<br>ix) Exercícios e seus acertos;<br>x) Prática final.                       |

## 8. Procedimentos didático-metodológicos

O curso é composto por dez módulos distintos nomeados segundo os componentes curriculares constantes na matriz curricular. Cada um dos módulos é constituído por uma série de até quatro videoaulas de quinze minutos cada, leituras orientadas na apostila didática, questionários aplicados diretamente no ambiente virtual de aprendizagem e laboratórios virtuais de aprendizagem com exercícios de programação interativos e autocorreção.

## 9. Descrição dos principais instrumentos de avaliação

A avaliação ao longo do curso se dará ao final de cada módulo – avaliação modular (AM) - de modo que o aluno só estará apto a avançar para o módulo seguinte a após a obtenção de aproveitamento igual ou superior a 70% na AM do atual.

As AM serão sempre compostas de um *quiz* sobre os conceitos estudados e de um

desafio de programação aplicados ao final do respectivo módulo, nas quais os discentes devem obter o aproveitamento mínimo necessário para a progressão. Ao longo do módulo, os laboratórios virtuais de aprendizagem, compostos por códigos desenvolvidos de maneira assistida, permitirão uma autoavaliação de seu desempenho propiciando parâmetros norteadores para a verificação continuada da aprendizagem.

O módulo final, consistirá de um projeto final de ferramenta computacional escolhido pelo discente dentre várias propostas. O código apresentado será analisado de maneira parametrizada (segundo barema de avaliação) por um *tester* automático configurado na sala virtual.

## 10. Definição dos mínimos de frequência e/ou aproveitamento da aprendizagem para fins de aprovação/certificação

A frequência mínima para aprovação será definida como sendo uma fração de  $\frac{3}{4}$  do número de dias úteis no prazo previsto de integralização, neste caso, três semanas (15 dias úteis), portanto, sendo necessários pelo menos 11 acessos à plataforma para obtenção da frequência mínima.

Para fins de certificação, o discente além de manter a frequência mínima exigida deverá obter rendimento global mínimo de 70%. O rendimento global será calculado pela média aritmética simples dos aproveitamentos de cada um dos módulos.

## 11. Infraestrutura física e equipamentos

O Instituto Federal de Minas Gerais, em seus variados campi, possui estúdios de EaD equipados com modernos sistemas de captação de vídeo e áudio, sistemas de iluminação e sistema de isolamento acústica.

Além disso, possui equipe técnica multidisciplinar que atua na definição de políticas e padrões para o ensino a distância, acompanhando as etapas de pré-produção, produção e pós-produção.

As videoaulas ficarão armazenadas em uma plataforma de streaming e as salas virtuais em servidores dedicados na reitoria da instituição, constantemente acompanhados por técnicos especializados.

Ainda se faz importante ressaltar que o Campus Avançado Piumhi conta com 2 laboratórios de informática equipados com 20 máquinas: laboratório I e 40 máquinas no laboratório II, além de os laboratórios possuem os requisitos necessários para desenvolver o curso de Python.

## 12. Referências

MicroPython. Particularidade da Linguagem de Programação. Disponível em: <https://micropython.org/>. Data do acesso: 06 de maio de 2020.

Python Brasil. **Conheça Python**. Disponível em: <https://python.org.br/>. Data do acesso: 25 de junho de 2020.

Python: O que é? Por que usar?. O que é Python?. Disponível em: <http://pyscience-brasil.wikidot.com/python:python-oq-e-pq>. Data do acesso: 25 de maio de 2020.

VIRTANEN, Pauli et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. **Nature methods**, v. 17, n. 3, p. 261-272, 2020.

## Anexo I – Plano de Ensino

Plano de ensino do curso proposto.

|   |                           |                          |
|---|---------------------------|--------------------------|
| <b>NOME DO CURSO:</b>   |                           |                          |
| <b>Programador Python: Nível Intermediário</b>  |                           |                          |
| <b>CH teórica:</b><br>20h   | <b>CH prática:</b><br>10h | <b>CH total:</b><br>30 h |
| <b>Ementa:</b> Iteradores, Recursividade, ordenação e busca, Funções Anônimas, Exemplos Práticos, <i>List comprehensions, mapping e filter</i> , Erros e Exceções, Manipulação de Arquivos, Exercícios e seus acertos.  |                           |                          |
| <b>Objetivos gerais:</b> O objetivo geral desse curso é proporcionar conhecimentos à programação de computadores na linguagem <i>Python</i> .   |                           |                          |
| <b>Objetivos específicos:</b> São objetivos específicos deste curso: <ol style="list-style-type: none"> <li>1. Capacitar o estudante para desenvolver programas básicos utilizando a linguagem de Programação <i>Python</i>, tendo condições e entendimento para continuar o desenvolvimento da linguagem.</li> <li>2. Contribuir para o processo de formação dos atores inscitos, possibilitando o desenvolvimento de programas em <i>Python</i>.</li> <li>3. Aplicar a linguagem de programação <i>Python</i> para resolução de problemas científicos e tecnológicos.</li> <li>4. Possibilitar a compreensão do o paradigma da programação orientada a objeto em <i>Python</i> utilizando a sintaxe da linguagem e características de organização de classes em pacotes e projetos na solução de problemas, assim como a implementação de softwares com recursos gráficos (GUI – <i>Graphical User Interface</i>).</li> </ol> |                           |                          |
| <b>Bibliografia Básica:</b> <ol style="list-style-type: none"> <li>1. MENEZES, Nilo Ney Coutinho. <b>Introdução à Programação com Python: Algoritmos e lógica de programação para iniciantes</b>. São Paulo: Novatec, 2017. 2ª edição.</li> <li>2. RAMALHO, Luciano. <b>Python Fluente</b>. São Paulo: Novatec, 2016.</li> <li>3. TUKER, Allen B. <b>[Programming languages: principles and paradigms. Português]</b>. São Paulo: McGraw-Hill, 2009.</li> <li>4. VIRTANEN, Pauli et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. <b>Nature methods</b>, v. 17, n. 3, p. 261-272, 2020.</li> </ol>  |                           |                          |
| <b>Bibliografia Complementar:</b> <ol style="list-style-type: none"> <li>1. MicroPython. <b>Particularidade da Linguagem de Programação</b>. Disponível em: <a href="https://micropython.org/">https://micropython.org/</a>. Data do acesso: 06 de maio de 2020.</li> <li>2. Python Brasil. <b>Conheça Python</b>. Disponível em: <a href="https://python.org.br/">https://python.org.br/</a>. Data do acesso: 25 de junho de 2020.</li> </ol>  |                           |                          |