# **Cálculo de Sub-redes**

**Bárbara Lauber - Erick Alair - Julia Stela - Otávio Nacke***Projeto realizado para a matéria de Redes de Computadores, acompanhado pelo professor Alexandre Yuji Kajihara*

## **Introdução**

Com a crescente demanda por recursos de internet e a expansão das redes, torna-se essencial gerenciar os endereços IP de forma eficiente. Subnetting é uma técnica fundamental para otimizar o uso do espaço de endereços IP e segmentar redes de maneira eficaz. Este relatório apresenta o desenvolvimento de um programa em Python que automatiza o processo de subnetting. O programa calcula e retorna o primeiro e o último endereço de cada sub-rede, bem como a máscara de sub-rede, com base no endereço IP, na máscara e na quantidade de sub-rede desejadas pelo usuário. A implementação desta ferramenta visa facilitar o trabalho dos administradores de rede, tornando o planejamento e a gestão de redes complexas mais simples e eficientes.

## **Objetivo**

O objetivo deste projeto é desenvolver um programa em Python que:

* Receba como entrada um endereço IP, uma máscara de sub-rede e a quantidade de sub-rede desejadas.
* Valide as entradas fornecidas pelo usuário.
* Calcule o primeiro e o último endereço de cada sub-rede, bem como a máscara de sub-rede.
* Exiba os resultados de forma clara e estruturada em uma interface gráfica.

## **Materiais**

* **Python 3.x**: Linguagem de programação utilizada para o desenvolvimento do programa.
* **Tkinter**: Biblioteca padrão do Python para a criação de interfaces gráficas.
* **Biblioteca ipaddress**: Utilizada para manipulação e cálculo de endereços IP.

## **Método**

O desenvolvimento do programa foi dividido em várias etapas, conforme detalhado a seguir:

* **Coleta de Dados do Usuário**:
  + Entrada do endereço IP.
  + Entrada da máscara de sub-rede (em formato CIDR).
  + Entrada da quantidade de sub-rede desejadas.
* **Validação dos Dados**:
  + Verificação da validade do endereço IP.
  + Verificação da validade da máscara de sub-rede.
  + Verificação se a quantidade de sub-rede é um número inteiro positivo.
* **Cálculo da Sub-rede**:
  + Determinação do número de bits necessários para a sub-rede com base na quantidade de sub-rede.
  + Cálculo da nova máscara de sub-rede.
  + Cálculo dos intervalos de endereços IP para cada sub-rede.
* **Geração dos Endereços de Sub-rede**:
  + Cálculo do primeiro endereço de cada sub-rede.
  + Cálculo do último endereço de cada sub-rede.
  + Retorno da máscara de sub-rede para cada sub-rede.
* **Exibição dos Resultados**:
  + Exibição do primeiro e do último endereço de cada sub-rede.
  + Exibição da máscara de sub-rede para cada sub-rede.

A implementação em Python seguiu estas etapas, utilizando a biblioteca **Tkinter** para a criação da interface gráfica e a biblioteca **ipaddress** para a manipulação dos endereços IP. O código fonte completo é apresentado na **main.py**.

## **Conclusão**

O desenvolvimento deste programa em Python proporcionou uma ferramenta eficiente para a geração e cálculo de sub-rede, facilitando o trabalho de administradores de rede ao permitir a visualização rápida e precisa dos intervalos de endereços IP e das máscaras de sub-rede. A interface gráfica desenvolvida com **Tkinter** oferece uma usabilidade simples e intuitiva, enquanto a utilização da biblioteca **ipaddress** garante a precisão dos cálculos. Futuras melhorias podem incluir a implementação de mais funcionalidades e a otimização do código para lidar com um número maior de sub-rede.

## **Referências**

* [Tkinter Documentation](https://docs.python.org/3/library/tkinter.html)
* [ipaddress — IPv4/IPv6 manipulation library](https://docs.python.org/3/library/ipaddress.html)
* [Math library](https://docs.python.org/3/library/math.html)
* Referências disponibilizadas no arquivo enviado no Moodle pelo professor Alexandre.