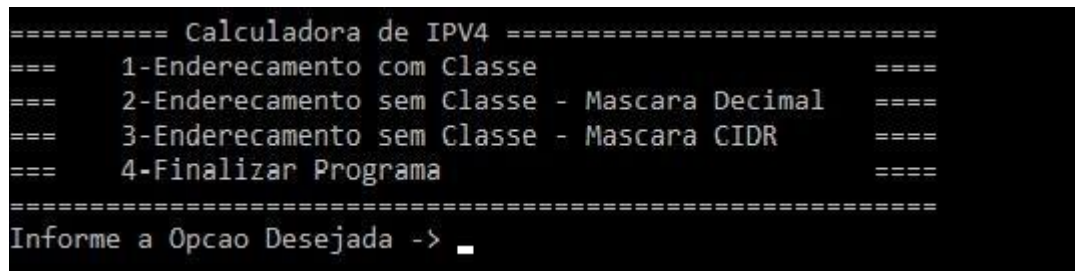


Memorial Descritivo

Nesse memorial será explicado a funcionalidade de um programa desenvolvido na linguagem C++, onde tem como objetivo ter a funcionalidade de uma calculadora de Ipv4, onde cada módulo do programa é necessário entrada de dados e através dessa entrada vamos obter uma saída esperada.

O Software AED_Redes foi desenvolvido utilizando a IDE DEV++, podendo ser executado em todas as plataformas Windows. O mesmo foi desenvolvido em modo texto e tem como tela de início as seguintes opções:

- 1-Endereçamento com Classe**
- 2-Endereçamento sem Classe - Mascara Decimal**
- 3-Endereçamento sem Classe - Mascara CIDR**
- 4-Finalizar Programa**



```
===== Calculadora de IPV4 =====  
== 1-Enderecamento com Classe ==  
== 2-Enderecamento sem Classe - Mascara Decimal ==  
== 3-Enderecamento sem Classe - Mascara CIDR ==  
== 4-Finalizar Programa ==  
===== Informar a Opcao Desejada -> _
```

Figura 1 – Menu De Opções

Cada opção tem como entrada um número de 1 a 4, números fora desse intervalo vai retornar uma mensagem de opção inválida e continuará no menu principal. Agora se tiver dentro do intervalo o software vai entrar nas funcionalidades que se encontram dentro da opção escolhida.

1- Endereçamento com Classe

Ao escolher essa opção o programa vai te pedir que informe um número de IP no formato (xxx.xxx.xxx.xxx) onde o caractere (.) é o delimitador de cada parte do IP digitado.

Ex: 10.125.10.2

```
===== Calculadora de IPV4 =====
=== 1-Endereçamento com Classe ===
=== 2-Endereçamento sem Classe - Mascara Decimal ===
=== 3-Endereçamento sem Classe - Mascara CIDR ===
=== 4-Finalizar Programa ===
=====
Informe a Opcao Desejada -> 1
Informe um Endereco de IP: 10.125.10.2
Informe a Classe do seu IP:
```

Figura 2 - Endereçamento com Classe

Após digitar o IP no padrão correto o mesmo vai passar por uma função de validação onde o mesmo vai ser analisado e verificado se realmente é um IP valido. A função se encontra no código do programa como seguinte nome: **IPValido**, sendo que recebe um IP e retorna **true** ou **false**.

```
bool IPValido (string ip){
```

Figura 3 – Função IPValido.

A função realiza a seguinte verificação:

- Verificar se o IP possui caracteres diferentes de números e o delimitador (.).
- Verifica se cada bloco do IP possui um número no intervalo de 0 a 255. - Verifica se existem mais de 3 delimitadores

Após essa verificação terá um retorno de **true** ou **false** como informado no parágrafo anterior deixando continuar o programa somente se o IP for válido. Como podemos notar na figura 2 o IP foi validado e logo em seguida foi pedido para o usuário informar a classe do IP.

OBS: Caso o usuário não saiba a classe do IP será retornado um erro informando que a classe está errada e logo em seguida um manual contendo os intervalos das classes.

Com a classe correta o programa continuará a sua linha de execução, onde terá com saída ao usuário as seguintes informações sendo elas na notação binária e decimal:

- **IP.**

- Máscara.
- Endereço De Rede.
- Endereço Broadcast.
- Número de Hosts.

```

Informe a Opcao Desejada -> 1
Informe um Endereco de IP: 10.125.10.2
Informe a Classe do seu IP: A

=====
                        Notacao Binaria
=====

IP.....: 00001010.01111101.00001010.00000010
Mascara.....: 11111111.00000000.00000000.00000000
Endereco De Rede.....: 00001010.00000000.00000000.00000000
Endereco De Broadcast.....: 00001010.11111111.11111111.11111111

=====

                        Notacao Decimal
=====

IP.....: 10.125.10.2
Mascara.....: 255.0.0.0
Endereco De Rede.....: 10.0.0.0
Endereco De Broadcast.....: 10.255.255.255
Numero De Hosts Por Rede.....: 16777216

=====

```

Figura 4 – Saída de dados da opção 1

Como podemos observar na Figura 4, após digitar a classe ele vai retornar como saída de dados os informados acima nas notações binária e decimal.

Para realizar essa operação foi utilizada 6 funções que vão ser descrita na página seguinte desse memorial.

Função ConversaoBinario:

```
string ConversaoBinario(string ip){
```

Figura 5 – Função ConversaoBinario.

Função desenvolvida para realizar a conversão do IP em binário, pois facilita as operações futuras do programa, como achar o Endereço de Rede e o Broadcast.

Função SeparadorDeBlocos:

```
vector<string> SeparadorDeBlocos(string ip){
```

Figura 6 – Função SeparadorDeBlocos.

Função desenvolvida para realizar a separação do IP em 4 blocos, ela consiste em realizar uma busca no IP e identificar o delimitador que no programa descrito é o caractere (.), quando identificado ele é retirado e o IP dividido em 4 blocos facilitando a conversão binária e diversas outras operações.

Função MascaraRede:

```
string MascaraRede (int cdr){
```

Figura 7 – Função MascaraRede.

Função desenvolvida para criar a máscara da rede, consiste em receber um CIDR e retornar a máscara, nessa opção de endereçamento com classe o CIDR é pré-definido de acordo com a classe que o IP pertence.

Função ConversaoDecimal:

```
string ConversaoDecimal(string x){
```

Figura 8 – Função ConversaoDecimal.

Como todas as operações devido a facilidade são realizadas em binário foi preciso desenvolver essa função para saída de dados em decimal, visto que é uma notação que todos os níveis de usuário conseguem compreender o que está descrito.

Função EnderecoDeRede:

```
string EnderecoDeRede(string ip,string mascara){
```

Figura 9 – Função EnderecoDeRede.

Função desenvolvida para realizar a operação logica AND com o IP e sua máscara, após essa operação ser feita é retornado o Endereço de Rede.

Função Broadcast:

```
string EnderecoBroadcast (string ip,string mascara){
```

Figura 10 – Função EnderecoBroadcast.

Função desenvolvida para realizar a operação logica OR com o IP e o complemento da máscara (Inverso), após essa operação ser feita é retornado o Endereço de Broadcast.

Com essas funções utilizadas o programa consegue retornar a resposta esperada, o número de host disponíveis que também se encontra na saída dessa opção é feita de maneira onde pegamos o número máximo de bits de um IP que é 32 Bits, realizando a operação de subtração com o CIDR, elevamos o resultado na base 2, com isso é obtido o número de hosts disponíveis na rede que o IP informado pertence. Lembre-se que nesse caso é considerado o endereço de Broadcast e o Endereço de Rede.

Valores mostrado ao usuário o programa retorna ao menu principal, onde o mesmo pode escolher a mesma ou outras opções disponíveis.

2-Endereçamento sem Classe - Máscara Decimal

Ao escolher essa opção o programa vai te pedir que informe um número de IP no formato (xxx.xxx.xxx.xxx) onde o caractere (.) é o delimitador de cada parte do IP digitado.

OBS: São realizadas as mesmas validações do IP que foi descrito nas páginas anteriores.

Ex: 192.168.41.1

```
===== Calculadora de IPV4 =====
=== 1-Endereçamento com Classe          ===
=== 2-Endereçamento sem Classe - Mascara Decimal ===
=== 3-Endereçamento sem Classe - Mascara CIDR    ===
=== 4-Finalizar Programa                ===
=====
Informe a Opcao Desejada -> 2
Informe um Endereco de IP: 192.168.41.1
Informe a Mascara desse IP: _
```

Figura 11 – Opção 2 do programa

Diferente da opção 1 desse menu, não será preciso informar a classe do IP, o que é pedido é a máscara em notação decimal a mesma só será aceita após passar por validações onde será feita análises na máscara informada como:

- Verificar se possui caracteres diferentes de números e o delimitador (.).
- Verifica se cada bloco possui um número no intervalo de 0 a 255.
- Verifica se existem mais de 3 delimitadores.
- Verifica se o primeiro Bloco é menor que 255.
- Verifica se a máscara em binário possui 0 no meio de bits 1.

Se a máscara passar na validação será mostrado na tela do usuário uma saída de dados que se encontra na figura 12 na página posterior.

```
Informe um Endereco de IP: 192.168.41.1
Informe a Mascara desse IP: 255.128.0.0

=====
                        Notacao Binaria
=====

IP.....: 11000000.10101000.00101001.00000001
Mascara.....: 11111111.10000000.00000000.00000000
Endereco De Rede.....: 11000000.10000000.00000000.00000000
Endereco De Broadcast.....: 11000000.11111111.11111111.11111111
Endereco IPV4 Inicial.....: 11000000.10000000.00000000.00000001
Endereco IPV4 Final.....: 11000000.11111111.11111111.11111110

=====

                        Notacao Decimal
=====

IP.....: 192.168.41.1
Mascara.....: 255.128.0.0
Endereco De Rede.....: 192.128.0.0
Endereco De Broadcast.....: 192.255.255.255
Endereco IPV4 Inicial.....: 192.128.0.1
Endereco IPV4 Final.....: 192.255.255.254
Numero De Hosts Por Rede.....: 8388606

=====
```

Figura 12 – Saída de dados opção 2

Além das funções já descrita nesse memorial foi utilizada mais duas funções onde a primeira consiste em achar o primeiro endereço válido dessa sub rede e a segunda função é utilizada para encontrar o ultimo endereço válido dessa sub rede, mas para ambas funcionar é preciso do endereço de rede e broadcast respectivamente.

OBS: Vale lembrar que todas as operações são realizadas em binário.

Como foi solicitado nos requisitos do software na opção 2, quando fosse ser informado o número de hosts disponíveis era para descontar o endereço de rede e o de broadcast, essa operação foi realizada e o valor mostrado ao usuário já se encontra com a retirada desses dois ips.

3-Endereçamento sem Classe - Mascara CIDR

Ao escolher essa opção o programa vai te pedir que informe um número de IP no formato (xxx.xxx.xxx.xxx) onde o caractere (.) é o delimitador de cada parte do IP digitado.

OBS: São realizadas as mesmas validações do IP que foi descrito nas páginas anteriores.

Ex: 198.145.10.2

```
=====
===== Calculadora de IPV4 =====
=== 1-Endereçamento com Classe ===
=== 2-Endereçamento sem Classe - Mascara Decimal ===
=== 3-Endereçamento sem Classe - Mascara CIDR ===
=== 4-Finalizar Programa ===
=====
Informe a Opcao Desejada -> 3
Informe um Endereco de IP: 198.145.10.2
Informe o CIDR (8 ate 32):
```

Figura 13 – Opção 3 do Programa.

Diferente da opção 1 desse menu, não será preciso informar a classe do IP, o que é pedido é a máscara em notação CIDR a mesma só será aceita se tiver entre o intervalo de 8 a 32.

A saída de dados consiste nas mesmas regras da opção 2, devido a isso não será necessário informar o que foi realizado pois já se encontra descrito em páginas acima, única mudança que foi feito é como a máscara foi informada, antes em notação decimal, agora sendo em notação CIDR.

Resultados do exemplo se encontram na figura 14 da página posterior em notação decimal e binaria.


```
Informe um Endereco de IP: 198.145.10.2
Informe o CIDR (8 ate 32): 27

=====
                        Notacao Binaria
=====

IP.....: 11000110.10010001.00001010.00000010
Mascara.....: 11111111.11111111.11111111.11100000
Endereco De Rede.....: 11000110.10010001.00001010.00000000
Endereco De Broadcast.....: 11000110.10010001.00001010.00011111
Endereco IPV4 Inicial.....: 11000110.10010001.00001010.00000001
Endereco IPV4 Final.....: 11000110.10010001.00001010.00011110

=====

=====
                        Notacao Decimal
=====

IP.....: 198.145.10.2
Mascara.....: 255.255.255.224
Endereco De Rede.....: 198.145.10.0
Endereco De Broadcast.....: 198.145.10.31
Endereco IPV4 Inicial.....: 198.145.10.1
Endereco IPV4 Final.....: 198.145.10.30
Numero De Hosts Por Rede.....: 30

=====
```

Figura 14 – Saída de dados opção 3

4-Finalizar Programa

Essa opção consiste em realizar o encerramento do executável do programa AED_Redes.exe. Após a escolha dessa opção o programa será finalizando, geralmente será escolhida quando o usuário tiver realizado todas as opções desejadas.