

Ronald Cassiano Da Silva Gonçalves

Ciência da computação

Conversão de Texto para Código Binário Utilizando Python

Resumo

Este artigo apresenta o desenvolvimento de um programa em Python capaz de converter palavras em código binário. O sistema solicita ao usuário uma letra e um conjunto de palavras, identifica aquelas que se iniciam com a letra informada e exibe suas representações binárias. A proposta visa demonstrar, de forma prática, conceitos fundamentais de programação estruturada, manipulação de strings e codificação de caracteres, destacando a importância desses conhecimentos na ciência da computação moderna.

1. Introdução

O avanço tecnológico e a crescente digitalização de processos tornam o domínio da lógica de programação e da codificação de dados um requisito essencial para profissionais da área de tecnologia da informação. Nesse contexto, compreender como os computadores interpretam e processam caracteres por meio de representações numéricas é um passo fundamental para o entendimento da base da computação.

O presente artigo aborda a criação de um programa simples em Python que converte texto em código binário. Tal aplicação, embora introdutória, reforça a importância dos sistemas de numeração binária, que constituem o alicerce das operações computacionais. Além disso, a proposta promove o aprendizado prático de estruturas de repetição, funções e manipulação de strings.

2. Fundamentação Teórica

A linguagem de programação Python foi escolhida por sua clareza sintática e ampla utilização no ensino de lógica e programação. Do ponto de vista conceitual, o programa explora os seguintes elementos fundamentais:

- Codificação de caracteres: cada caractere possui um valor numérico na tabela Unicode. A função `ord()` em Python permite obter esse valor.
- Representação binária: o formato `:08b` converte o número inteiro retornado por `ord()` em

uma sequência binária de 8 bits.

- Manipulação de strings: o método `.startswith()` é usado para verificar se uma palavra começa com uma determinada letra.
- Funções e modularização: o código é estruturado em funções (`texto_para_binario` e `main`) para favorecer a clareza e a reutilização.

3. Metodologia

O desenvolvimento do programa foi dividido em etapas sucessivas:

1. Definição da função de conversão.
2. Coleta e tratamento das entradas.
3. Processamento e filtragem.
4. Conversão e exibição dos resultados.

O código desenvolvido é apresentado a seguir:

```
def texto_para_binario(texto):
    return " ".join(f"{ord(c):08b}" for c in texto)

def main():
    letra = input("Digite uma letra: ").strip().lower()
    palavras = input("Digite palavras separadas por espaço: ").split()

    print(f"\nPalavras que começam com {letra}:\n")

    for palavra in palavras:
        if palavra.lower().startswith(letra):
            bin_palavra = texto_para_binario(palavra)
            print(f"Palavra: {palavra} | Binário: {bin_palavra}")

if __name__ == "__main__":
    main()
```

4. Resultados

Durante a execução do programa, o usuário insere uma letra e uma lista de palavras. O sistema identifica as que começam com a letra informada e exibe suas representações em binário. Exemplo de execução:

Digite uma letra: c

Digite palavras separadas por espaço: casa carro bola cachorro

Palavras que começam com 'c':

Palavra: casa | Binário: 01100011 01100001 01110011 01100001

Palavra: carro | Binário: 01100011 01100001 01110010 01110010 01101111

Palavra: cachorro | Binário: 01100011 01100001 01100011 01101000 01101111
01110010 01110010 01101111

5. Discussão

O projeto, embora simples, evidencia como pequenos algoritmos podem demonstrar conceitos fundamentais da computação. A transformação de texto em código binário é uma representação direta do modo como os computadores interpretam dados textuais. Além disso, o uso de funções e laços de repetição reforça práticas de programação estruturada.

6. Conclusão

O desenvolvimento do programa demonstrou que é possível, com poucos comandos, criar uma aplicação capaz de realizar operações significativas de conversão e análise de texto. O projeto contribui para o aprendizado prático de conceitos básicos de programação e ilustra, de forma concreta, a lógica de funcionamento dos sistemas computacionais binários.

Referências

PYTHON SOFTWARE FOUNDATION. Python 3 Documentation. Disponível em:

<https://docs.python.org/3/>

TANENBAUM, A. S. Organização Estruturada de Computadores. 6^a ed. Pearson, 2013.

SEBESTA, R. W. Conceitos de Linguagens de Programação. 11^a ed. Pearson, 2018.