

Universidade São Judas Tadeu - Butantã
Noturno

Nomes: Gabriel Carvalho dos Santos - 821159957
Gustavo Mendes de Lima - 821139542
Matheus dos Santos de Souza - 821133576

Turma: GQS-CCP1AN-BUE1

Professor: Robson Calvetti

```
def
    contar_palavras(texto):
        palavras = texto.split()
        return len(palavras)

def main():
    texto = input("Digite um texto: ")
    total_palavras = contar_palavras(texto)
    print(f"O total de palavras no texto é: {total_palavras}")

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

Plano de teste-

1. Teste com texto vazio

- Entrada: ""
- Saída Esperada: 0
- Descrição: Verificar se o software retorna 0 para um texto vazio.

2. Teste com texto sem palavras

- Entrada: " "
- Saída Esperada: 0
- Descrição: Verificar se o software conta corretamente espaços em branco.

3. Teste com uma única palavra

- Entrada: "Olá"
- Saída Esperada: 1
- Descrição: Verificar se o software conta uma palavra.

4. Teste com várias palavras

- Entrada: "Olá, como você está?"
- Saída Esperada: 5
- Descrição: Verificar se o software conta corretamente palavras em uma frase.

5. Teste com pontuação

- Entrada: "Olá! Você, quem?"
- Saída Esperada: 4
- Descrição: Verificar se o software conta palavras corretamente, ignorando pontuações.

6. Teste com múltiplos espaços

- Entrada: "Olá mundo"
- Saída Esperada: 2
- Descrição: Verificar se o software lida com múltiplos espaços entre palavras.

7. Teste com texto longo

- Entrada: "Este é um teste para contar a quantidade de palavras em um texto maior."
- Saída Esperada: 15
- Descrição: Verificar se o software funciona corretamente com um texto maior.

8. Teste com caracteres especiais

- Entrada: "#Python é @incrível!"
- Saída Esperada: 4
- Descrição: Verificar se o software conta corretamente palavras com caracteres especiais.

Método de Teste

1. Execute o software e insira as entradas conforme os casos de teste.
2. Verifique se a saída do programa corresponde à saída esperada.
3. Registre os resultados e anote quaisquer discrepâncias.

Resultado Esperado

Todos os testes devem passar, indicando que a função de contagem de palavras está funcionando corretamente.

CASOS DE TESTES -

Caso de Teste 1: Texto Vazio

- Descrição: Testar a contagem de palavras em um texto vazio.
- Entrada: ""
- Saída Esperada: 0
- Resultado: [Passou/Falhou]

Caso de Teste 2: Texto com Espaços em Branco

- Descrição: Testar a contagem de palavras em uma string que contém apenas espaços.
- Entrada: " "

- Saída Esperada: 0
- Resultado: [Passou/Falhou]

Caso de Teste 3: Uma Única Palavra

- Descrição: Testar a contagem de uma única palavra.
- Entrada: "Olá"
- Saída Esperada: 1
- Resultado: [Passou/Falhou]

Caso de Teste 4: Frase Com Várias Palavras

- Descrição: Testar a contagem em uma frase normal.
- Entrada: "Olá, como você está?"
- Saída Esperada: 5
- Resultado: [Passou/Falhou]

Caso de Teste 5: Texto com Pontuação

- Descrição: Testar a contagem de palavras que incluem pontuação.
- Entrada: "Olá! Você, quem?"
- Saída Esperada: 4
- Resultado: [Passou/Falhou]

Caso de Teste 6: Múltiplos Espaços Entre Palavras

- Descrição: Verificar a contagem com espaços múltiplos entre palavras.
- Entrada: "Olá mundo"
- Saída Esperada: 2
- Resultado: [Passou/Falhou]

Caso de Teste 7: Texto Longo

- Descrição: Testar a contagem em um texto maior.
- Entrada: "Este é um teste para contar a quantidade de palavras em um texto maior."

- Saída Esperada: 15
- Resultado: [Passou/Falhou]

Caso de Teste 8: Caracteres Especiais

- Descrição: Testar a contagem com caracteres especiais.
- Entrada: "#Python é @incrível!"
- Saída Esperada: 4
- Resultado: [Passou/Falhou]

Registro de Resultados

- Data: [Data do Teste]
- Executor do Teste: [Nome do Testador]
- Comentários: [Notas adicionais sobre o teste, se necessário]

ROTEIRO DE TESTES -

Introdução

- Objetivo do Software:
 - Contar a quantidade de palavras em um texto fornecido pelo usuário.
- Público-alvo:
 - Estudantes, escritores e qualquer pessoa que precise contar palavras em textos.

2. Pré-requisitos

- Ambiente:
 - Python instalado no computador.
- Editor:
 - Qualquer editor de texto (Ex: VSCode, PyCharm, Notepad++).

3. Instruções de Instalação

1. Copiar o Código:

- O usuário deve copiar o código fornecido para um arquivo com a extensão `.py`.

2. Salvar o Arquivo:

- Salve o arquivo com um nome descritivo, como `contar_palavras.py`.

4. Execução do Software

1. Abrir o Terminal:

- No Windows, use o CMD ou PowerShell.
- No Mac ou Linux, use o Terminal.

2. Navegar até o Diretório:

- Use o comando `cd` para navegar até o diretório onde o arquivo foi salvo.

3. Executar o Programa:

- Digite o comando: `python contar_palavras.py` e pressione Enter.

5. Interação com o Usuário

- Entrada:
 - O programa solicitará que o usuário digite um texto.
- Exemplo de Mensagem:
 - "Digite um texto:"
- Processamento:
 - O programa contará as palavras no texto fornecido.

6. Saída do Programa

- Resultado:
 - O programa exibirá o total de palavras contadas.
- Exemplo de Mensagem:
 - "O total de palavras no texto é: X"

7. Exemplos de Uso

- Exemplo 1:

- Entrada: "Olá, mundo!"
- Saída: "O total de palavras no texto é: 2"
- **Exemplo 2:**
 - Entrada: ""
 - Saída: "O total de palavras no texto é: 0"

8. Conclusão

- Finalização:
 - O usuário pode encerrar o programa após visualizar a contagem de palavras.
- Sugestões para Melhorias Futuras:
 - Adicionar suporte para contar frases.
 - Implementar uma interface gráfica (GUI).
 - Adicionar opções para salvar resultados em um arquivo.

9. Suporte

- Contato:
 - Incluir informações de contato para dúvidas ou feedback.