

Sumário

1. Introdução:	2
2. Implementação:	2
3. Testes	2
4. Conclusão	3
Referências	3
Anexos	3
NomeArquivo.c	3

1. Introdução

Este trabalho tem como objetivo a adaptação do algoritmo Quicksort em linguagem C para ordenar um vetor de palavras. O vetor de exemplo contém 20 palavras. O programa realizará a ordenação em ordem crescente com base no tamanho das palavras, contará o número de trocas e comparações realizadas, calculará a mediana do conjunto de dados e registrará todas essas informações em um arquivo CSV.

GitHub:

<https://github.com/Matheus-Inacioal/TP02/tree/main>

2. Implementação:

A implementação deste trabalho utiliza a linguagem de programação C. Foram criadas funções comparar para a ordenação, trocar para trocar elementos de um vetor de ponteiros e quicksort para realizar a ordenação personalizada. O programa principal lê um vetor de 20 palavras, ordena-o com base no tamanho das palavras usando o algoritmo Quicksort personalizado e registra as trocas, comparações e mediana em um arquivo CSV chamado "saida.csv"

3. Testes

As Palavras ficaram fora de ordem

```
put > saída.txt
1  abacate
2  abacaxi
3  amora
4  banana
5  caqui
6  cereja
7  figo
8  framboesa
9  goiaba
10 kiwi
11 laranja
12 limão
13 maca
14 manga
15 melancia
16 morango
17 papaya
18 pera
19 pêssego
20 uva
```

4. Conclusão

Neste trabalho, implementamos com sucesso um programa em C que utiliza o algoritmo Quicksort para ordenar um vetor de palavras com base no tamanho das palavras. Foram registradas as trocas, comparações e mediana em um arquivo CSV. Os testes realizados garantiram a correção do programa. Como melhoria, poderíamos implementar uma solução que permitisse ao usuário fornecer seu próprio vetor de palavras.

Referências

FEILOFOFF, Paulo. Algoritmos em Estruturas de Dados: Aulas. Disponível em: <https://www.ime.usp.br/~pf/algoritmos/aulas/quick.html>. [18 de setembro de 2023].

Anexos

saida.csv

NomeArquivo.c

código-fonte.c

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
int comparar(const void *a, const void *b) {  
    return strlen(*(const char **)a) - strlen(*(const char **)b);  
}
```

```
void trocar(char **a, char **b) {  
    char *temp = *a;  
    *a = *b;  
    *b = temp;  
}
```

```
void quicksort(char **arr, int esquerda, int direita, int *trocas, int *comparacoes) {  
  
    if (esquerda < direita) {  
  
        int i = esquerda, j = direita;  
  
        char *pivot = arr[(i + j) / 2];  
  
        while (i <= j) {  
  
            while (strlen(arr[i]) < strlen(pivot)) {  
  
                i++;  
  
                (*comparacoes)++;  
  
            }  
  
            while (strlen(arr[j]) > strlen(pivot)) {  
  
                j--;  
  
                (*comparacoes)++;  
  
            }  
  
            if (i <= j) {  
  
                trocar(&arr[i], &arr[j]);  
  
                i++;  
  
                j--;  
  
                (*trocas)++;  
  
            }  
  
        }  
  
        quicksort(arr, esquerda, j, trocas, comparacoes);  
  
        quicksort(arr, i, direita, trocas, comparacoes);  
    }  
}
```

```
}  
  
}  
  
int main() {  
  
    char *arr[20] = {  
  
        "maca", "banana", "pera", "uva", "laranja", "abacaxi", "limao", "manga", "abacate", "kiwi",  
  
        "cereja", "morango", "pessego", "goiaba", "melancia", "framboesa", "amora", "caqui", "figo", "papaya"  
  
    };  
  
  
    int trocas = 0, comparacoes = 0;  
  
  
    quicksort(arr, 0, 19, &trocas, &comparacoes);  
  
  
    char *mediana = arr[9];  
  
  
  
    FILE *arquivoCSV = fopen("saida.csv", "w");  
    if (arquivoCSV == NULL) {  
  
        perror("Erro ao abrir o arquivo de saída");  
  
        return 1;  
  
    }  
  
  
  
    fprintf(arquivoCSV, "Palavra, Tamanho\n");  
  
  
  
    for (int i = 0; i < 20; i++) {  
  
        fprintf(arquivoCSV, "%s, %d\n", arr[i], (int)strlen(arr[i]));  
  
    }  
}
```

```
}

fprintf(arquivoCSV, "Trocas: %d\n", trocas);

fprintf(arquivoCSV, "Comparacoes: %d\n", comparacoes);

fprintf(arquivoCSV, "Mediana: %s\n", mediana);


fclose(arquivoCSV);


printf("Vetor ordenado por tamanho das palavras foi salvo em 'saida.csv'.\n");

printf("Número de trocas: %d\n", trocas);

printf("Número de comparações: %d\n", comparacoes);

printf("Mediana: %s\n", mediana);


return 0;

}
```