

**Universidade de Brasília**

**Departamento de Ciência da Computação**



## **Lista de Exercício 3 - OA**

### **Autores:**

Gabriel Bessa 16/0120811

Thiago Veras 16/0146682

### **Disciplina:**

Organização de Arquivos

### **Turma:**

A

### **Professor:**

Oscar Gaidos

Brasília  
12 de Abril de 2018

## Exercícios:

1- Faça um programa que leia os dados sobrenome, nome, endereço, CEP e telefone do teclado e grave esses dados segundo os métodos abaixo. Esse programa deve ser capaz de listar os registros após gravá-los:

```
#####  
#                                     #  
#          FILE CONCATENATOR          #  
#                                     #  
#####  
#                                     #  
#      Aluno: Thiago Veras Machado    #  
#           M: 16/0146682             #  
# Aluno: Gabriel Cunha Bessa Vieira  #  
#           M: 16/0120811             #  
#                                     #  
#####
```

Figura 1: Entrada do programa

```
Nome: Gabriel  
Sobrenome: Bessa  
Endereço: SQN 412 Bloco A Apto 108  
CEP: 70867010  
Telefone: 981235281
```

Figura 2: Entrada do programa

## Explicação do Código:

```
1 #include <stdio.h>    // Use function Printf()  
2 #include <iostream>   // Use cin and cout functions  
3 #include <fstream>    // Use open() function  
4 #include <regex>      // Use vector class  
5 #include <string>     // Use String to read files name  
6  
7 // Use collors on printf  
8 #define RED          "\x1b[31m"  
9 #define GREEN        "\x1b[32m"  
10 #define CYAN         "\x1b[36m"  
11 #define RESET        "\x1b[0m"  
12  
13 // Used to omit ::std syntax  
14 using namespace std;
```

Começo do código com todas as bibliotecas para uso de funções e defines para cores no terminal.

```
1 vector<string> message = {"Nome: ", "Sobrenome: ", "  
    Endereco: ", "CEP: ", "Telefone: "}, data(5);  
2  
3  
4 void intro(){  
5     printf(CYAN"\n");  
6     printf("#####\n");  
7     printf("#                               #\n");  
8     printf("#           FILE CONCATENATOR           #\n");  
9     printf("#                               #\n");  
10    printf("#####\n");  
11    printf("#                               #\n");  
12    printf("#       Aluno: Thiago Veras Machado       #\n");  
13    printf("#           M: 16/0146682               #\n");  
14    printf("# Aluno: Gabriel Cunha Bessa Vieira #\n");  
15    printf("#           M: 16/0120811               #\n");  
16    printf("#                               #\n");  
17    printf("#####\n");  
18    printf("\n" RESET);  
19 }
```

Introdução do arquivo no terminal.

```
1 int main() {  
2  
3     // Display intro message  
4     intro();  
5  
6     for(int i = 0; i < 5; i++){  
7         cout << message[i];  
8         getline(cin,data[i]);  
9     }  
10    save_by_fixed_size();  
11    save_by_begin_size();  
12    save_by_separator();  
13    save_by_valor();  
14 }
```

Função principal do código, que faz o mesmo rodar

a) Campos de tamanho fixo.

```
1 void save_by_fixed_size(){
2     fstream arquivo;
3     arquivo.open("fixed_size.txt", ios::app | ios::out);
4     for(int i = 0; i < 5; i++){
5         for(int j = 0; j < 20; j++){
6             if(j >= data[i].size()) arquivo << '-';
7             else arquivo << data[i][j];
8         }
9     }
10    arquivo << '\n';
11    arquivo.close();
12 }
```

Essa função salva os valores fornecidos em um arquivo .txt com o nome de: "*fixed\_size.txt*".



Figura 3: Arquivo texto de Fixed size

b) Campos com tamanho no início.

```
1 void save_by_begin_size(){
2     fstream arquivo;
3     arquivo.open("begin_size.txt", ios::app | ios::out);
4     for(int i = 0; i < data.size(); i++) arquivo << data[i]
5         ].size() << data[i];
6     arquivo << '\n';
7     arquivo.close();
8 }
```

Essa função salva os valores fornecidos em um arquivo .txt com o nome de: "*begin\_size.txt*".



Figura 4: Arquivo texto de Begin size

c) Campos com separadores.

```

1 void save_by_separator(){
2     fstream arquivo;
3     arquivo.open("separator.txt", ios::app | ios::out);
4     for(int i = 0; i < data.size(); i++) arquivo << data[
        i] << '%';
5     arquivo << '\n';
6     arquivo.close();
7 }

```

Essa função salva os valores fornecidos em um arquivo .txt com o nome de: "*separator.txt*".

Gabriel%Bessa%SQN 412 Bloco A Apto 108%70867010%981235281%

Figura 5: Arquivo texto de Separator

d) Campos com atributo = valor.

```

1 void save_by_valor(){
2     fstream arquivo;
3     arquivo.open("valor.txt", ios::app | ios::out);
4     for(int i = 0; i < data.size(); i++) arquivo <<
        message[i] << data[i] << "|";
5     arquivo << '\n';
6     arquivo.close();
7 }

```

Essa função salva os valores fornecidos em um arquivo .txt com o nome de: "*valor.txt*".

Nome: Gabriel|Sobrenome: Bessa|Endereço: SQN 412 Bloco A Apto 108|CEP: 70867010|Telefone: 981235281|

Figura 6: Arquivo texto de Valor