

Lista de Exercícios 1

Alunos: Giovanni Guidini 16/0122660; Vitor Dullens 16/0148260.

Prof. Oscar Gaidos

Organização de Arquivos

CIC 116327

Data: 29/03/2018

Exercicio 1

Implemente um programa que leia e concatene UM OU MAIS arquivos textos e grave o arquivo concatenado em DISCO ou o exibe no VÍDEO. Faça esse programa o mais seguro possível, isto é, após operações de E/S, verifique se não houve erro. Se houve, sinalize para o usuário e contorne a situação. O programa deverá ter a opção de ler caracter a caracter ou leitura de bloco.

Explicação

Nosso código realiza uma operação com um número indeterminado de arquivos, operando em um arquivo por vez. Ao final de cada operação é possível mostrar os resultados, adicionar mais um arquivo à concatenação, ou salvar os resultados em um outro arquivo (que já exista ou não).

Código

```
1  /* Estudantes:
   *      Giovanni Guidini      16/0122660;
   *      Vitor Fernandes Dullens 16/0148260;
   *      Disciplina:
   *      Organizacao de Arquivos 1/2018
   *
   *      Compiled using g++ (GCC) 7.3.1 20180312
   *      with flags -std=c++11 -Wall
   *      in Linux 4.14.30-1-MANJARO
   */
11
12 #include <stdio.h>
13 #include <iostream> //comandos de E/S (cin, cout)
14 #include <fstream> //ler arquivos externos
15 #include <string> //trabalhar com strings
16
17 //define, para limpar a tela em diferentes sistemas operacionais
18 #ifdef WIN32
19     #define CLEAR "cls"
20 #else
21     #define CLEAR "clear"
22 #endif
23
24 // define utilizacao do namespace
25 using namespace std;
26
27 void menu() { //menu inicial
28     system(CLEAR);
29     printf("=====\\n");
30     printf(" Concatenador de Arquivos      \\n");
31     printf("=====\\n");
32
33     printf("\\n\\n(press ENTER to continue)\\n");
```

```

35 }
36
37 int type(){ //como que o usuario deseja concatenar os arquivos
38     int type;
39
40     printf("Concatenar:\n");
41     printf("\t1 - linha por linha\n");
42     printf("\t2 - palavra por palavra\n");
43     printf("\t3 - char por char\n");
44     printf("\t4 - arquivo por arquivo\n");
45     printf("opcao: ");
46     scanf("%d", &type);
47     getchar();
48
49     while(type < 0 or type > 4){ //tratamento de erro
50         printf("—— opcao invalida, escreva uma opcao valida ---\n");
51         printf("opcao: ");
52         scanf("%d", &type);
53         getchar();
54     }
55     return type;
56 }
57
58 string readFile(fstream file, int mode){ //ler arquivo dependendo do comando
59     escolhido na funcao type()
60     int n;
61     if(mode == 1){
62         printf("Ler quantas linhas?\n");
63         cin >> n;
64     }
65     else if(mode == 2){
66         printf("Ler quantas palavras?\n");
67         cin >> n;
68     }
69     else if(mode == 3){
70         printf("Ler quantos caracteres?\n");
71         cin >> n;
72     }
73     else{
74         // arquivo inteiro == numero enorme de linhas
75         n = 100000;
76     }
77
78     string r = "", line;
79     if(mode == 1 || mode == 4){
80         int i = 0;
81         while(getline(file, line) && i < n){
82             r += line + "\n"; // getline function removes \n from line
83             i++;
84         }
85     }
86     else if(mode == 2){
87         int i = 0;
88         while(i < n){
89             file >> line;
90             r += line + " ";
91             i++;
92         }
93     }
94 }

```

```

93     else{
94         char read[n+1];
95         file.get(read, n+1);
96         r += string(read);
97     }
98
99     return r;
100 }
101
102 fstream openFile(string file){ // abrir arquivo
103     fstream arquivo (file , ios::in);
104     if(!arquivo){ //tratamento de erro
105         cout << "Failed to open\n";
106         return (fstream) NULL;
107     }
108
109     return arquivo;
110 }
111
112 string fileOP(){ //manuseio de arquivo
113     string file;
114     system(CLEAR);
115     printf("Qual o nome do arquivo?\n");
116     cin >> file;
117     fstream ini = openFile(file);
118     if(!ini){ //tratamento de erro
119         string again;
120         printf("Tentar novamente? [Y/n]\n");
121         cin >> again;
122         if(again == "Y" || again == "y")
123             return fileOP();
124         else
125             return "";
126     }
127
128     int t = type();
129     return readFile(openFile(file), t);
130 }
131
132 int continua(){ //segundo menu de opcoes
133     printf("\nO que voce quer fazer agora?\n");
134     printf("\t(1) - Adicionar outro arquivo\n");
135     printf("\t(2) - Mostrar resultados\n");
136     printf("\t(3) - Finalizar operacao\n");
137
138     int type;
139     scanf("%d", &type);
140     getchar();
141     while(type < 0 or type > 3){ //tratamento de erro
142         printf("—— opcao invalida, escreva uma opcao valida ——\n");
143         printf("opcao: ");
144         scanf("%d", &type);
145         getchar();
146     }
147     return type;
148 }
149
150 int main(){
151     menu();
152     string file;

```

```

153     string content = "";
154     // arquivo inicial
155     content += fileOP();

156
157     // menu das operacoes
158     int go = continua();
159     while(go != 3){
160         if(go == 1){
161             content += fileOP();
162         }
163         else{
164             system(CLEAR);
165             cout << content;

166
167             printf("\n\n(press ENTER to continue)\n");
168             getchar();
169         }
170         go = continua();
171     }

172
173     printf("Qual arquivo voce quer salvar?\n");
174     cin >> file;
175     fstream f (file , ios::out | ios::app);
176     f << content;
177     return 0;
}

```

ex1.cpp

Saídas

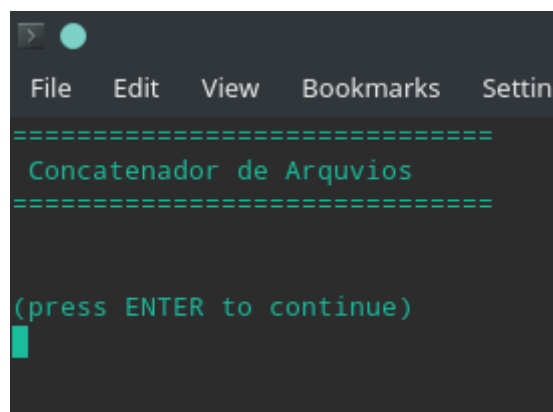


Figura 1: Tela de abertura

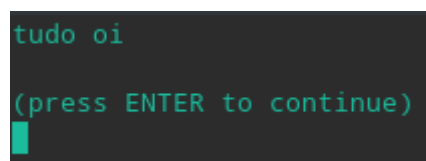


Figura 2: Resultado de concatenar por palavras

```
Qual o nome do arquivo?  
ex1.cpp  
Concatenar:  
    1 - linha por linha  
    2 - palavra por palavra  
    3 - char por char  
    4 - arquivo por arquivo  
opcao: 1  
Ler quantas linhas?  
5  
  
O que voce quer fazer agora?  
    (1) - Adicionar outro arquivo  
    (2) - Mostrar resultados  
    (3) - Finalizar operação
```

Figura 3: Sequência típica de operações

```
A long time ago, in a galaxy far, far away  
  
- May the Force be...  
  
- with you  
  
:)  
  
(press ENTER to continue)
```

Figura 4: Resultado de concatenar por arquivos

Exercicio 2

Pesquise e compare as diferenças entre um arquivo texto e um arquivo binário, do ponto de vista físico.

Resposta

Ambas contem dados armazenados, porém existe diferença na codificação destes dados, os Arquivos textos seguem codificação ASCII ou UTF-8, por exemplo, enquanto os binários n seguem nenhuma codificação específica. Com isso, no arquivo binário, cada byte representa um dado, enquanto no arquivo texto, cada byte representa um caracter.

Normalmente um arquivo texto é um arquivo que pode ser lido e compreendido por um ser humano e aberto em um editor de texto comum. Já um arquivo binário é todo o resto, algo feito para o computador entender, como arquivos executáveis e imagens.