

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <stdnoreturn.h>
5
6 /*|Matheus Martins Batista (2019005687) - Sistemas Operacionais (COM120) |
7 |Ciências da Computação (CCO) - EP12 - Exercício01 - 05/12/2021          |*/
8
9 //Definir os structs utilizados ao decorrer do código
10 typedef struct cluster {
11
12     char dados[512];
13     unsigned long int prox;
14
15 } cluster;
16
17 int v_clusters[100];
18
19 typedef struct data {
20
21     unsigned short dia;
22     unsigned short mes;
23     unsigned short ano;
24     unsigned short hora;
25     unsigned short min;
26     unsigned short seg;
27
28 } data;
29
30 typedef struct arquivos {
31
32     char nome[8]; //nome do arquivo iniciado com o hexadecimal 0ebh são arquivos deletados
33     char extensao[3];
34     // Proteção
35     unsigned short sistema; //arquivo do sistema
36     unsigned short hidden; //arquivo oculto
37     unsigned short archived; //arquivo arquivado
38     // Data e hora da criação
39     data criacao;
40     // Data do último acesso
41     data acesso;
42     // Tamanho
43     unsigned long int tamanho;
44     // Cluster inicial
45     unsigned long int cluster;
46
47 } arquivo;
48
49 int main() {
50     cluster ctr;
51     FILE *arq;
52     arquivo a, arq_ext4;
53     int qtd_cluster = 0;
54     arq = fopen("Disco.dat", "rb");
55
56     //Buscar apenas o arquivo EXT4.TXT no Disco.dat
57     while (!feof(arq)) {
58         fflush(stdin);
59         fread(&a, sizeof (arquivo), 1, arq);
60
61         if (!strcmp(a.nome, "EXT4") && !strcmp(a.extensao, "TXT")) {

```

```

62         arq_ext4 = a;
63         break;
64     }
65 }
66
67 //Registrar o tamanho do arquivo EXT4.TXT e o cluster atual
68 unsigned long int cluster_atual = arq_ext4.cluster;
69 unsigned long int tam_arq = arq_ext4.tamanho;
70
71 while (!feof(arq)) {
72     fseek(arq, sizeof (cluster) * cluster_atual, SEEK_SET);
73     fread(&ctr, sizeof (cluster), 1, arq);
74     v_clusters[qtd_cluster] = cluster_atual;
75
76     if (ctr.prox) {
77         cluster_atual = ctr.prox;
78         qtd_cluster++;
79         printf("%s", ctr.dados);
80         tam_arq -= 512;
81
82     } else {
83         qtd_cluster++;
84
85         for (int i = 0; i < tam_arq; i++) {
86             printf("%c", ctr.dados[i]);
87         }
88         break;
89     }
90 }
91
92 printf("\nQuantidade de clusters: %d\n", qtd_cluster);
93 printf("Tamanho do arquivo: %lu\n", arq_ext4.tamanho);
94 fclose(arq);
95 }

```