

```

1  #define ex2
2  #include "stdio.h"
3
4  #ifndef ex1
5  /*
6  1 - Faça um programa para:
7      1 - criar/abrir um arquivo texto de nome "arq.txt",
8      2 - receba via teclado diversos caracteres (um por vez)e armazene nesse
9      arquivo. O caracter '0' finaliza a entrada de dados.
10     3 - reabra o arquivo fazendo o ponteiro apontar para seu início
11     4 - leia e imprima na tela todos os caracteres armazenados.
12  */
13  main(){
14  //FILE *fopen(); /*a funcao fopen() retorna um ponteiro do tipo "FILE". Por isso,
15  //      tem que fazer esta declaracao */
16  FILE *p; /*declara o ponteiro "p" do tipo "FILE"*/
17  char ch;
18  char i;
19  int n=0;
20
21  if((p = fopen("arq.txt","w"))==NULL)
22  {
23      printf("erro na abertura do arquivo \n");
24      exit(0); /*encerra o programa e volta ao sistema operacional*/
25  }
26  do{
27      printf("Digite um caracter\n");
28      scanf("%c", &ch);
29      getchar();
30      putc(ch,p);
31
32      n++;
33  }while(ch != '0');
34
35  fclose(p);
36
37  p = fopen("arq.txt","r");
38  char c;
39  for(n; n > 0; n-- ){
40      c =getc(p);
41      if(c != '0'){
42          putchar(c);
43      }
44  }
45  fclose(p);
46  }
47
48  #endif // ex1
49
50  #ifndef ex2
51  /*
52  2 - Faça um programa para gerenciar uma agenda de contatos. Para cada contato
53      armazene o nome, o telefone e o aniversário (dia e mês). O programa deve ter
54      as opções:
55      1 - inserir contato,
56      2 - remover contato,
57      3 - pesquisar um contato pelo nome,
58      4 - listar todos os contatos,
59      5 - listar os contatos cujo nome inicia com uma dada letra,
60      6 - imprimir os aniversariantes do mês.
61      Sempre que o programa for encerrado, os contatos devem ser armazenados. Quando
62      o programa iniciar, os contatos devem ser inicializados com os dados contidos
63      no arquivo.
64  */
65
66  struct contatos{
67      char nome[255];
68      char telefone[12];
69      char dataNas[5];
70  };
71  void listaContato(struct contatos *c, FILE *p);
72  void insereContato(struct contatos *c, FILE *p);
73  main(){
74
75  FILE *p;
76  struct contatos contato;
77  struct contatos *c;
78  int fim =0;
79  c = &contato;
80
81  if((p = fopen("contato.txt","w"))==NULL)
82  {
83      printf("erro na abertura do arquivo \n");
84      exit(0); /*encerra o programa e volta ao sistema operacional*/

```

```

85     }
86     do{
87
88         printf("MENU: \n");
89         printf("1 - INSERIR\n2 - REMOVER\n3 - PESQUISAR\n4 - LISTAR TODOS\n5 - LISTAR POR
INICIAL\n6 - ANIVERSARIANTES DO MES\n7 - FINALIZAR\n\n");
90         scanf("%d", &fim);
91
92         if(fim == 1){
93             listaContato(c, p);
94             insereContato(c, p);
95         }
96
97
98
99         fclose(p);
100
101     }while(fim != 7);
102
103     }
104     //inserir contato
105     void insereContato(struct contatos *c, FILE *p){
106         printf("Digite o nome: ");
107         scanf("%s", c->nome);
108         fprintf(p, "%s", c->nome);
109         fflush(stdin);
110         printf("Digite o telefone: ");
111         scanf("%s", c->telefone);
112         fprintf(p, "%s", c->telefone);
113         fflush(stdin);
114         printf("Digite a data nascimento dia e mes (dd/mm): ");
115         scanf("%s", c->dataNas);
116         fprintf(p, "%s", c->dataNas);
117         fflush(stdin);
118     }
119
120     // listar todos os contatos
121     void listaContato(struct contatos *c, FILE *p){
122         if((p = fopen("contato.txt", "r"))== NULL)
123         {
124             printf("erro na abertura do arquivo \n");
125             exit(0);
126         }
127         printf("lista contatos\n");
128         fscanf(p, "%s", c->nome);
129         fprintf(p, "%s", c->nome);
130         printf("Nome: %s\n", c->nome );
131         fscanf(p, "%s", c->telefone);
132         fprintf(p, "%s", c->telefone);
133         printf("Telefone: %s\n", c->telefone );
134         fscanf(p, "%s", c->dataNas);
135         fprintf(p, "%s", c->dataNas);
136         printf("Data Nascimento: %s\n", c->dataNas );
137     }
138
139
140     #endif // ex2
141

```