

```

1  #define ex6
2  #include "stdio.h"
3
4  #ifndef ex1
5  /*1 - Escreva um programa que receba via teclado um char, int, long, unsigned,
6      float, double, unsigned long e unsigned char, declare ponteiros para os
7      mesmos e imprima-os no video utilizando o operador de conteudo * no
8      seguinte formato:
9
10         10         20         30         40         50         60
11         123456789012345678901234567890123456789012345678901234567890
12         int         long         double         unsigned         char
13
14         float         unsigned long         unsigned char*/
15 main()
16 {
17     int i, *pi;
18     char c, *pc;
19     long l, *pl;
20     unsigned int u, *pu;
21     float f, *pf;
22     double d, *pd;
23     unsigned long ul, *pul;
24     unsigned char uc, *puc;
25
26     printf("Digite um char: ");
27     scanf("%c", &c);
28     fflush(stdin);
29     printf("Digite um int: ");
30     scanf("%d", &i);
31     fflush(stdin);
32     printf("Digite um long: ");
33     scanf("%ld", &l);
34     fflush(stdin);
35     printf("Digite um unsigned int: ");
36     scanf("%u", &u);
37     fflush(stdin);
38     printf("Digite um float: ");
39     scanf("%f", &f);
40     fflush(stdin);
41     printf("Digite um double: ");
42     scanf("%ld", &d);
43     fflush(stdin);
44     printf("Digite um unsigned long: ");
45     scanf("%lu", &ul);
46     fflush(stdin);
47     printf("Digite um unsigned char: ");
48     scanf("%c", &uc);
49     fflush(stdin);
50
51     pi = &i;
52     pc = &c;
53     pl = &l;
54     pu = &u;
55     pf = &f;
56     pd = &d;
57     pul = &ul;
58     puc = &uc;
59
60     printf("         10         20         30         40         50         60\n");
61     printf("123456789012345678901234567890123456789012345678901234567890\n");
62
63     printf("         %-6d", *pi);
64     printf("         %-11lu", *pl);
65     printf("         %-5u\n", *pu);
66     printf("         %-8.1e", *pf);
67     printf("         %-11ld", *pd);
68     printf("         %c\n", *pc);
69     printf("         %-10lu", *pul);
70     printf("         %u\n", *puc);
71 }
72 #endif // ex1
73
74
75
76 #ifndef ex2
77 /*2 - Com a estrutura abaixo, defina um vetor de estruturas de 4 elementos.
78     Receba os dados pelo teclado usando ponteiros e imprima-os no video
79     também usando ponteiros. Utilize um comando de loop. (vetor de estruturas)
80         nome, end, cidade, estado, cep
81 */
82 struct pessoas{
83     char nome[50];
84     char end[50];

```

```

85     char cidade[50];
86     char estado[3];
87     char cep[10];
88 };
89 main(){
90     struct pessoas pessoa[4];
91     struct pessoas *ps;
92     int i=0;
93     ps = pessoa;
94
95     for(i = 0; i < 4; i++){
96         printf("Digite o %d nome: ", i);
97         gets((ps+i)->nome);
98         printf("Digite o %d endereco: ", i);
99         gets((ps+i)->end);
100        printf("Digite o %d cidade: ", i);
101        gets((ps+i)->cidade);
102        printf("Digite o %d estado: ", i);
103        gets((ps+i)->estado);
104        printf("Digite o %d cep: ", i);
105        gets((ps+i)->cep);
106    }
107    printf("\nListagem da estrutura:\n\n");
108    for(i = 0; i < 4; i++){
109        printf("nome = %s\n", (ps+i)->nome);
110        printf("endereco = %s\n", (ps+i)->end);
111        printf("cidade = %s\n", (ps+i)->cidade);
112        printf("estado = %s\n", (ps+i)->estado);
113        printf("cep = %s\n\n", (ps+i)->cep);
114    }
115 }
116 #endif // ex2
117
118
119 #ifdef ex3
120 /*3 - Receba um caracter via teclado usando ponteiros e compara com o vetor abaixo
121 usando ponteiros. mostre como resultado se são IGUAIS ou DIFERENTES.
122 vetor -> b,d,f,h,j,l,m,o,q,s,u,w,y
123 */
124
125 main(){
126     int i=0, verifica = 0;
127     char *p, a, *pvet;
128     char vet [] = "bdfhjlmogsuwy";
129
130     p = &a;
131     pvvet = vet;
132
133     printf("Digite um char: ");
134     scanf("%c", p);
135
136     for(i=0; *(pvvet + i) != '\0'; i++){
137         if(*p == *(pvvet + i)){
138             verifica++;
139         }
140     }
141
142     if(verifica > 0){
143         printf("SAO IGUAIS");
144     }else{
145         printf("SAO DIFERENTES");
146     }
147 }
148 #endif // ex3
149
150
151 #ifdef ex4
152 /*4 - Receba 2 string de ate 10 caracteres via teclado, compare-as usando ponteiros
153 e mostre como resultado se são IGUAIS ou DIFERENTES.
154 */
155
156 //funcao valida string
157 int validaString(char *ps1, char *ps2){
158     int i;
159     int ver=0;
160
161     for(i=0; i<10;i++){
162         if(*(ps1+i) == '\0' && *(ps2+i) != '\0'){
163             ver ++;
164             break;
165         }else
166         if(*(ps1+i) != '\0' && *(ps2+i) == '\0'){
167             ver ++;
168             break;

```

```

169     }else
170     if(*(ps1+i) == '\0' && *(ps2+i) == '\0'){
171         break;
172     }
173     if(*(ps1+i) != *(ps2+i)){
174         ver++;
175         break;
176     }
177 }
178 return ver;
179 }
180
181 main(){
182
183     char str1[10],str2[10], *ps1, *ps2;
184     int valida=0;
185     int i=0;
186     ps1 = str1;
187     ps2 = str2;
188
189     printf("Digite 2 strings de ate 10 caracteres\n\n");
190
191     printf("Digite a 1 string: ");
192     gets(str1);
193
194     printf("\nDigite a 2 string: ");
195     gets(str2);
196
197     valida = validaString(ps1, ps2);
198
199     if(valida > 0){
200         printf("AS STRINGS SAO DIFERENTES\n");
201     }else{
202         printf("AS STRINGS SAO IGUAIS\n");
203     }
204 }
205
206 #endif // ex4
207
208
209
210
211 #ifdef ex5
212 /*
213 5 - Escreva um programa que receba via teclado numeros inteiros positivos usando
214 ponteiros. Quando o numero digitado for negativo o programa deve parar e calcula
215 a media dos valores digitados e mostrar o resultado.
216 */
217 main(){
218     int n, *pn, soma=0, i=-1;
219     float media;
220     pn = &n;
221     do{
222         printf("Digite um numero: ");
223         scanf("%d", pn);
224         if(*pn > 0){
225             soma += *pn;
226         }
227         i++;
228     }while(*pn > 0);
229     media = (float)soma / i;
230     printf("A media foi %.2f", media);
231 }
232 #endif // ex5
233
234
235 #ifdef ex6
236 /*6 - Escreva um programa que contenha uma estrutura representando uma data
237 valida. Essa estrutura deve conter os campos dia, mes e ano. Em seguida,
238 leia duas datas e armazene nessa estrutura usando ponteiros. Calcule e exiba o
239 numero de dias que decorrem entre as duas datas usando ponteiros.*/
240
241 typedef struct {
242     int dia;
243     int mes;
244     int ano;
245 } Data;
246
247 int bissexto (int ano);
248 unsigned long dist_dias (Data inicio, Data fim);
249
250 /* guarda o numero de dias em cada mes para anos normais e bissextos */
251 int dias_mes[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},
252                        {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

```

```

253
254 int main(void) {
255     Data dia1, dia2, *pd1, *pd2;
256
257     pd1 = &dia1;
258     pd2 = &dia2;
259
260     printf("Coloque data inicial no formato: dia/mes/ano\n");
261     scanf("%d/%d/%d", &(*pd1).dia, &(*pd1).mes, &(*pd1).ano);
262     printf("Coloque data final no formato: dia/mes/ano\n");
263     scanf("%d/%d/%d", &(*pd2).dia, &(*pd2).mes, &(*pd2).ano);
264
265     printf("A distancia em dias: %lu\n", dist_dias (*pd1, *pd2));
266
267     return 0;
268 }
269
270 // Retorna 1 caso 'ano' seja bissexto, 0 caso contrário
271
272 int bissexto (int ano) {
273     return (ano % 4 == 0) && ((ano % 100 != 0) || (ano % 400 == 0));
274 }
275
276 // Retorna a distancia entre inicio e fim em dias.
277 // Assume que inicio nao vem depois de fim.
278
279 unsigned long dist_dias (Data pd1, Data pd2) {
280     unsigned long idias, fdias; /* guarda qtos dias tem da data */
281                             /* ate o comeco do ano */
282     unsigned long def_anos = 0; /* guarda diferenca entre anos das */
283                             /* datas inicio e fim medida em dias */
284     register int i;
285     int dbissexto;
286
287     idias = pd1.dia;
288     dbissexto = bissexto (pd1.ano);
289     for (i = pd1.mes - 1; i > 0; --i)
290         idias += dias_mes[dbissexto][i];
291
292     fdias = pd2.dia;
293     dbissexto = bissexto (pd2.ano);
294     for (i = pd2.mes - 1; i > 0; --i)
295         fdias += dias_mes[dbissexto][i];
296
297     while (pd1.ano < pd2.ano)
298         def_anos += 365 + bissexto(pd1.ano++);
299
300     return def_anos - idias + fdias;
301 }
302
303 #endif // ex6
304

```