54struct.c

```
 1 /* 54struct.c トランプの53枚のカードを表現する */
 2 #include <stdio.h>
 3 #include <stdlib.h>
 4 #include <string.h>
 5 #include <time.h>
 6
 7 #define SPADE 0
 8 #define HEART 1
 9 #define CLOVER 2
10 #define DIA 3
11 #define JOKER -1
12
13 #define OMOTE 1
14 #define URA 0
15
16 #define MAX 53
17
18 /* トランプ構造体 */
19 struct card{
20         char type;      /* 0:スペード 1:ハート 2:クラブ 3:ダイヤ -1:ジョーカー */
21         char no;        /* 1～13  ジョーカー:0 */
22         char face;      /* 0: 裏 1:表 */
23 };
24
25 /* ----- カードを表示する関数 */
26 void showCard(struct card x){
27         if(x.face == URA){      /* 裏なら表示しない */
28                 printf("X¥n");
29                 exit(1);
30         }
31         switch(x.type){
32                 case SPADE:
33                         printf("S");
34                         break;
35                 case HEART:
36                         printf("H");
37                         break;
38                 case CLOVER:
39                         printf("C");
40                         break;
41                 case DIA:
42                         printf("D");
43                         break;
44                 default:
45                         printf("Joker¥n");
46         }
47         /* ジョーカー以外は数値も表示 */
48         if(x.no) printf("%3d¥n", x.no);
49 }
50
```

```
51 /* ----- メイン関数 */
52 int main(void)
53 {
54     int i, j;
55     struct card c[MAX];     /* 4種類×12枚＋ジョーカー1枚 = 53枚 */
56     struct card work;       /* シャッフルしたカード退避用 */
57     int r1, r2;             /* シャッフル用 */
58     /* カードを作る（ジョーカー以外52枚） */
59     for(i=0; i<4; i++){
60         for(j=0; j<13; j++){
61             c[i*13+j].type = i;
62             c[i*13+j].no = j+1;
63             c[i*13+j].face = URA;
64         }
65     }
66     /* カードを作る（ジョーカー） */
67     c[MAX-1].type = -1;
68     c[MAX-1].no = 0;
69     c[MAX-1].face = URA;
70     /* カードをシャッフルする */
71     srand((unsigned)time(NULL));
72     for(i=0; i<100; i++){
73         r1 = rand() % MAX;
74         r2 = rand() % MAX;
75         /* カードをシャッフル */
76         work.type = c[r1].type;
77         work.no = c[r1].no;
78         work.face = c[r1].face;
79
80         c[r1].type = c[r2].type;
81         c[r1].no = c[r2].no;
82         c[r1].face = c[r2].face;
83
84         c[r2].type = work.type;
85         c[r2].no = work.no;
86         c[r2].face = work.face;
87     }
88
89     /* 上から7枚表示してみる */
90     for(i=0; i<7; i++){
91         c[i].face = OMOTE;     /* 表にして */
92         showCard(c[i]);        /* 表示 */
93     }
94
95     return 0;
96 }
97 /* 改良：シャッフル部分をコンパクトに */
```