

CHAPTER 7

구조체와 포인터 그리고  
함수 공용체와 열거형 실습

# 실습 1 문제

문제 1)

int, double 변수를 구조체와 공용체로 만들어서 각각 출력

```
16  
8
```

# 실습 1 정답

```
1  #include<stdio.h>
2
3  typedef struct structBox
4  {
5      int member1;
6      int member2;
7      double member3;
8  } SBox;
9
10 typedef union unionBox
11 {
12     int member1;
13     int member2;
14     double member3;
15 } UBox;
16
17 int main(void)
18 {
19     SBox sbox;
20     UBox ubox;
21     printf("%d \n", sizeof(SBox));
22     printf("%d \n", sizeof(UBox));
23     return 0;
24 }
```

16  
8

## 실습 2 문제

문제 2)

Typedef 로 열거형 각 요일을 선언한다.

이때 요일의 매칭되는 숫자는 일~토 : 1~7 로 순서대로 지정

```
SUN : 1
MON : 2
TUE : 3
WED : 4
THU : 5
FRI : 6
SUN : 7
```

## 실습 2 정답

```
1 #include <stdio.h>
2
3 typedef enum week{
4     SUN=1, MON=2, TUE=3, WED=4, THU=5, FRI=6, SAT=7
5 }WEEK;
6
7 int main(void){
8     WEEK sun = SUN;
9     WEEK mon = MON;
10    WEEK tue = TUE;
11    WEEK wed = WED;
12    WEEK thu = THU;
13    WEEK fri = FRI;
14    WEEK sat = SAT;
15
16    printf("SUN : %d \n", sun);
17    printf("MON : %d \n", mon);
18    printf("TUE : %d \n", tue);
19    printf("WED : %d \n", wed);
20    printf("THU : %d \n", thu);
21    printf("FRI : %d \n", fri);
22    printf("SAT : %d \n", sat);
23
24    return 0;
25 }
```

```
SUN : 1
MON : 2
TUE : 3
WED : 4
THU : 5
FRI : 6
SAT : 7
```

## 실습 3 문제

문제 3)

3인 블랙잭 카드 게임 프로그램 구현  
dealer(컴퓨터), player1, player2

블랙잭 카드 게임 참고 링크

<https://blog.naver.com/yjhjr8/221761487128>

<https://blog.naver.com/ban2yrabbit/221973301777>

카드를 표현하는 자료구조를 만들고,  
초기 카드 무늬/숫자 정보로 카드를 초기화한다.

카드를 섞는 함수를 만든다.

카드 게임을 진행하면서 필요한 다양한 기능(카드 분배, 금액 베팅, ..)을  
함수로 구성한다.

카드 게임 승자를 판단하고, 보유 금액을 수정하는 기능을 함수로 구현한다.

## 실습 3 문제

문제 3)

열거형으로 shape (spade, club, dia, heart) 및  
num (ace=1,jack=11,queen12, king=13 선언)

결과 후 "계속하시겠습니까?" 출력에서 y 또는 Y 입력시 게임 계속 진행  
다른 입력시 게임 종료

# 실습 3 문제

블랙잭 게임에 오신 것을 환영합니다! 지금 바로 시작하시겠습니까? [y/n] y

-----  
| ♣     |  
| 4     |  
♣

score : 4

<<PLAYER1>>

CASH : 500

-----  
| ♣     |  
| 5     |  
♣

score : 5

<<PLAYER2>>

CASH : 500|

-----  
| ♠     |  
| 5     |  
♠

score : 5

PLAYER1님, 돈을 걸어봅시다! 500 중 얼마를 배팅하시겠습니까? 200

PLAYER2님, 돈을 걸어봅시다! 500 중 얼마를 배팅하시겠습니까? 100



# 실습 3 문제

```
#####TURN : PLAYER1 님#####

HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: h
-----
|♦   |
| 2  |
|   ♦|
-----
score : 7
HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: h
-----
|♣   |
| 7  |
|   ♣|
-----
score : 14
HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: s
#####TURN : PLAYER2 님#####

HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: s
#####DEALER's TURN#####

-----
|♠   |
| 10 |
|   ♠|
-----
score : 14
-----
|♠   |
| 6  |
|   ♠|
-----
score : 20

게임 결과>> DEALER: 20 / PLAYER1: 14 / PLAYER2: 5
prize: 310

결과>> P1 CASH : 300 / P2 CASH : 400
계속하시겠습니까? y
```

# 실습 3 정답

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <math.h>
4 #include <time.h>
5 #include <string.h>
6
7 #define AI 17
8 #define SPADE "♠"
9 #define CLUB "♣"
10 #define DIA "♦"
11 #define HEART "♥"
12
13 // enum
14 enum shape {spade, club, dia, heart};
15 enum num { ace = 1, jack = 11, queen = 12, king = 13 };
16
17 // 카드 구조체
18 struct Card {
19     int number;
20     char shape;
21 };
22
23 // 모든 카드
24 struct Card card_all[52];
25 int next;
26
27 // 선수
28 struct Player {
29     int cash;
30     int score;
31     struct Card card_player[21];
32 };
33
34 struct Player player[3];
```

## 실습 3 정답

```
36 // 카드 섞는 함수
37 void shuffle()
38 {
39     srand(time(NULL));
40     for (int i = 0; i < 52; i++)
41     {
42         int index = rand() % 52;
43         struct Card temp = card_all[index];
44         card_all[index] = card_all[i];
45         card_all[i] = temp;
46     }
47 }
48
49 // 카드 채우는 함수
50 void filldeck()
51 {
52     int i = 0;
53     // 4개의 모양과 13개의 숫자를 짝지어서 카드 만듦
54     for (int j = 0; j < 4; j++)
55     {
56         for (int k = 1; k < 14; k++)
57         {
58             card_all[i].number = k;
59             card_all[i].shape = j;
60             i++;
61         }
62     }
63 }
64
65 // 섞은 후 앞에서 부터 한장 나눠주는 함수
66 struct Card deal(int next)
67 {
68     return card_all[next];
69 }
```

# 실습 3 정답

```
71 // 사용자가 뽑은 카드를 문자열이 아닌 카드 모양으로 출력, 점수 계산
72 void printcard(int p, int turns)
73 {
74     int ace_score=0;
75
76     printf("-----\n");
77
78     switch(player[p].card_player[turns].shape)
79     {
80     case spade:
81         printf("|%s    |\n", SPADE);
82         break;
83     case club:
84         printf("|%s    |\n", CLUB);
85         break;
86     case dia:
87         printf("|%s    |\n", DIA);
88         break;
89     case heart:
90         printf("|%s    |\n", HEART);
91         break;
92     }
93
94     switch(player[p].card_player[turns].number)
95     {
96     case ace:
97         printf("|  A  |\n");
98         ace_score = -1;
99         break;
100    case jack:
101        printf("|  J  |\n");
102        player[p].score += 10;
103        break;
104    case queen:
105        printf("|  Q  |\n");
106        player[p].score += 10;
107        break;
108    case king:
109        printf("|  K  |\n");
110        player[p].score += 10;
111        break;
112    default:
113        printf("| %2d  |\n", player[p].card_player[turns].number);
114        player[p].score += player[p].card_player[turns].number;
115        break;
116    }
```

# 실습 3 정답

```
118     switch(player[p].card_player[turns].shape)
119     {
120     case spade:
121         printf("|    %s|\n", SPADE);
122         break;
123     case club:
124         printf("|    %s|\n", CLUB);
125         break;
126     case dia:
127         printf("|    %s|\n", DIA);
128         break;
129     case heart:
130         printf("|    %s|\n", HEART);
131         break;
132     }
133
134     printf("-----\n");
135
136     // ace 계산 하기
137     if(ace_score == -1)
138     {
139         printf("ACE가 나왔네요, [1/11]점으로 계산합니다: ");
140         scanf("%d", &ace_score);
141         player[p].score += ace_score;
142     }
143
144     printf("score : %d\n", player[p].score);
145
146     return;
147 }
148
149 // 배팅하는 함수
150 int betting(struct Player player)
151 {
152     int bet;
153
154     printf("돈을 걸어봅시다! %d 중 얼마를 배팅하시겠습니까? ", player.cash);
155     scanf("%d", &bet);
156
157     if(bet > player.cash)
158     {
159         printf("이런이런... 돈이 부족한군요. 조금만 아껴야겠어요. 다시 입력 : ");
160         scanf("%d", &bet);
161         return bet;
162     }
163     else return bet;
164 }
```

# 실습 3 정답

```
166 // 게임을 시작
167 void staylorhit(int p, int turns)
168 {
169     char answer;
170
171     printf("\n#####TURN : PLAYER%d님#####\n\n",p);
172     while(player[p].score < 21)
173     {
174         printf("HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: ");
175         scanf(" %c", &answer);
176
177         if (answer == 'h' || answer == 'H')
178         {
179             turns++;
180             player[p].card_player[turns] = deal(next++);
181             printcard(p, turns);
182         }
183         else if (answer == 's' || answer == 'S')
184         {
185             break;
186         }
187         else
188         {
189             printf("HIT이면 h, STAY면 s라니까요.\n다시,");
190         }
191     }
192
193     if(player[p].score>21)
194     {
195         printf("21점을 넘었습니다. LOSS\n");
196         player[p].score = 0;
197     }
198     else if (player[p].score==21)
199     {
200         printf("21점입니다. WIN\n");
201     }
202 }
```

# 실습 3 정답

```
204 // 딜러는 지능적으로 16까지만 안전하게 여기고 hit
205 void dealer(int turns)
206 {
207     printf("\n#####DEALER's TURN#####\n\n");
208     while(player[0].score < 21)
209     {
210         if (player[0].score < AI)
211         {
212             turns++;
213             player[0].card_player[turns] = deal(next++);
214             printcard(0, turns);
215         }
216         else
217         {
218             break;
219         }
220     }
221     if(player[0].score>21) player[0].score = 0;
222 }
223
224 // 게임 진행
225 void play()
226 {
227     int turns = 0; int goon = 1; int push = 0; int push_flag = 0; int bet_total = 0;
228
229     while(goon)
230     {
231         int winner, winnerscore=-1;
232         int bet[3]; bet[0] = 100;
233         push_flag = 0;
234
235         if (push == 0) bet_total = 0;
236
237         // 두명 다 확인
238         if(player[1].cash<=0)
239         {
240             printf("!!! PLAYER1 파산하셨습니다. 100 다시 충전\n\n");
241             player[1].cash += 100;
242         }
243         else if(player[2].cash<=0)
244         {
245             printf("!!! PLAYER2 파산하셨습니다.\n\n");
246             player[2].cash += 100;
247         }
248         else if (player[1].cash<=0 && player[2].cash<=0)
249         {
250             printf("!!! PLAYER1 파산하셨습니다.\n\n");
251             printf("!!! PLAYER2 파산하셨습니다.\n\n");
252             break;
253         }
254     }
```

## 실습 3 정답

```
255 player[0].score = 0;
256 player[1].score = 0;
257 player[2].score = 0;
258
259 player[0].card_player[turns] = deal(next++);
260 printcard(0, turns);
261
262 printf("\n<<PLAYER1>>\n");
263 printf("CASH : %d\n", player[1].cash);
264 player[1].card_player[turns] = deal(next++);
265 printcard(1, turns);
266
267 printf("\n<<PLAYER2>>\n");
268 printf("CASH : %d\n", player[2].cash);
269 player[2].card_player[turns] = deal(next++);
270 printcard(2, turns);
271
272 bet_total += bet[0];
273 printf("\nPLAYER1님, ");
274 bet[1] = betting(player[1]);
275 bet_total += bet[1];
276 printf("\nPLAYER2님, ");
277 bet[2] = betting(player[2]);
278 bet_total += bet[2];
279
280 stayorhit(1, turns);
281 stayorhit(2, turns);
282
283 dealer(turns);
284
285 printf("\n게임 결과>> DEALER: %d / PLAYER1: %d / PLAYER2: %d\n", player[0].score, player[1].score, player[2].score);
```



# 실습 3 정답

```
288     for (int i=0; i<3; i++)
289     {
290         if (winnerscore < player[i].score)
291         {
292             winner = i;
293             winnerscore = player[i].score;
294         }
295     }
296     printf("prize: ");
297
298     for (int i=0; i<3; i++)
299     {
300         if (winnerscore == player[i].score && winner != i)
301         {
302             if (push == 0) push = 1;
303             push_flag = 1;
304         }
305     }
306
307     push *= push_flag;
308
309     if(push == 1)
310     {
311         player[1].cash -= bet[1];
312         player[2].cash -= bet[2];
313         printf("0\n");
314     }
315     else if(winner == 0)
316     {
317         player[1].cash -= bet[1];
318         player[2].cash -= bet[2];
319         push = 0;
320         printf("%d\n", bet_total);
321     }
322     else if(winner == 1)
323     {
324         player[1].cash += bet_total;
325         player[2].cash -= bet[2];
326         push = 0;
327         printf("%d\n", bet_total);
328     }
329     else if(winner == 2)
330     {
331         player[1].cash -= bet[1];
332         player[2].cash += bet_total;
333         push = 0;
334         printf("%d\n", bet_total);
335     }
```

## 실습 3 정답

```
337     printf("\n결과>> P1 CASH : %d / P2 CASH : %d\n", player[1].cash, player[2].cash);
338
339     char answer;
340     printf("계속하시겠습니까? ");
341     scanf(" %c", &answer);
342
343     if (answer=='y' || answer=='Y')
344     {
345         continue;
346     }
347     else
348     {
349         break;
350     }
351 }
352
353 printf("게임이 종료됩니다");
354 }
355
356 void reset()
357 {
358     player[0].score = 0; player[1].score = 0; player[2].score = 0;
359     player[1].cash = 500; player[2].cash = 500;
360
361     filldeck(); shuffle(); next = 0;
362 }
363
364 // 시작
365 void start()
366 {
367     char choice;
368     printf("블랙잭 게임에 오신 것을 환영합니다! 지금 바로 시작하시겠습니까? [y/n] ");
369     scanf("%c", &choice);
370 }
```

## 실습 3 정답

```
371 while(choice!='Y' && choice!='y' && choice!='N' && choice!='n')
372 {
373     printf("이상한 문자를 입력하셨습니다. 다시 입력해 주세요! [y/n] ");
374     scanf("%c", &choice);
375 }
376
377 if(choice=='Y' || choice=='y')
378 {
379     reset();
380     play();
381 }
382 else if(choice=='N' || choice=='n')
383 {
384     printf("게임을 종료합니다.\n");
385 }
386 return;
387 }
388
389 int main(void)
390 {
391     start();
392
393     return 0;
394 }
```

# 실습 3 정답

블랙잭 게임에 오신 것을 환영합니다! 지금 바로 시작하시겠습니까? [y/n] y

-----  
| ♣     |  
| 4     |  
♣

score : 4

<<PLAYER1>>

CASH : 500

-----  
| ♣     |  
| 5     |  
♣

score : 5

<<PLAYER2>>

CASH : 500|

-----  
| ♠     |  
| 5     |  
♠

score : 5

PLAYER1님, 돈을 걸어봅시다! 500 중 얼마를 배팅하시겠습니까? 200

PLAYER2님, 돈을 걸어봅시다! 500 중 얼마를 배팅하시겠습니까? 100

# 실습 3 정답

```
#####TURN : PLAYER1 님#####

HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: h
-----
|♦   |
| 2  |
|   ♦|
-----
score : 7
HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: h
-----
|♣   |
| 7  |
|   ♣|
-----
score : 14
HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: s
#####TURN : PLAYER2 님#####

HIT을 원하시면 h를, STAY를 원하시면 s를 입력하세요[h/s]: s
#####DEALER's TURN#####

-----
|♠   |
| 10 |
|   ♠|
-----
score : 14
-----
|♠   |
| 6  |
|   ♠|
-----
score : 20

게임 결과>> DEALER: 20 / PLAYER1: 14 / PLAYER2: 5
prize: 310

결과>> P1 CASH : 300 / P2 CASH : 400
계속하시겠습니까? y
```

# 질문 및 정리

