

課題 p13-1.c

構造体の宣言方法と使用方法、各要素に変数名を付けられることが分かりました。また、構造体の中の要素に変数名を付けて文字列を代入するときは、strcpy 関数を使用する必要があることが分かりました。

課題 p13-2.c

型の再定義の方法と、構造体で配列を宣言することで、配列の中にデータを代入された構造体を作ることができると分かりました。また、構造体を変数に代入して利用できることも理解できました。

課題 p13-3.c

アロー演算子で構造体の要素を参照、代入する方法と、構造体の要素のアドレスを記憶させて、使用メモリを削減する方法が分かりました。

課題 p13-4.c

構造体を自作関数に引き渡して、参照、代入する方法と、返し値を構造体にする方法が分かりました。

## 問題作成課題 (p13-5.c)

### プログラムソースコード画像

( プログラムが長くなるが小さくしすぎないこと。 )

```
1  /* p13-5.c */
2  #include<stdio.h>
3
4  struct hms {
5      int hour, min, sec;
6  };
7
8  int total_sec(struct hms tm) {
9      tm.min += tm.hour * 60;
10     tm.sec += tm.min * 60;
11
12     return tm.sec;
13 }
14
15 void addXsec(struct hms *p, int x) {
16     p->sec = total_sec(*p) + x;
17     p->hour = (p->sec / 3600);
18     p->min = (p->sec % 3600) / 60;
19     p->sec = (p->sec % 3600) % 60;
20
21     return ;
22 }
23
24 void printhms(struct hms tm) {
25     printf("%02d:%02d:%02d\n", tm.hour, tm.min, tm.sec);
26
27     return;
28 }
29
30 int main() {
31
32     struct hms game = { 3, 32, 10 };
33     struct hms dinner = { 1, 2, 33 };
34     int addtime;
35
36     printhms(game);
37     printf(" -> %d sec\n", total_sec(game));
38
39     addtime = 520;
40     addXsec(&game, addtime);
41     printhms(game);
42
43     printhms(dinner);
44     printf(" -> %d sec\n", total_sec(dinner));
45
46     addtime = -432;
47     addXsec(&dinner, addtime);
48     printhms(dinner);
49
50     return 0;
51 }
52
```

実行結果画像

```
N:\prog\prog13>gcc p13-5.c  
  
N:\prog\prog13>a  
03:32:10  
-> 12730 sec  
03:40:50  
01:02:33  
-> 3753 sec  
00:55:21
```

本日の感想や反省

構造体の宣言方法、ポインタ等での参照方法、関数内での利用方法を理解することができました。プログラムをかなり短くできるので、活用したいと思える仕様でした。