

構造体

型の違った（同じでもよい）ものをひとまとめにした型を定義する。

例 体重，年齢，血液型をひとまとめにした構造体'person'を定義する。

- 型の定義

```
struct person
{
    float weight;
    int    age;
    char  blood;
};
```

- 変数の宣言

```
struct person koizumi, takenaka, fukuda;
```

- 変数の参照

```
koizumi.weight=60; fukuda.age=62; takenaka.blood='A';
```

- ポインタを使った参照

構造体へのポインタを p とすると，

(*p). 項目名

とするか，あるいは

p->項目名

とする。

プログラム例

```
1 /* jikan.c (時間の計算) */
2 struct time
3 {
4     int hour;
5     int min;
6     int sec;
7 };
8
9 struct time henkan(int d)
10 /* 秒を何時何分何秒という形式に変更する */
11 {
12     struct time e;
13     int x;
14
15     x=d;
16     e.sec =x%60; x/=60;
17     e.min =x%60; x/=60;
18     e.hour=x%60;
19     return(e);
20 }
21
22 void printtime(struct time t)
23 {
24     printf(" %d 時間 %d 分 %d 秒 \n",t.hour,t.min,t.sec);
25 }
```

```
26
27 int byou(struct time h)
28 /* 何時何分何秒という形式を秒に変更する */
29 {
30     int s;
31     s=h.hour*3600+h.min*60+h.sec;
32     return(s);
33 }
34
35 main()
36 {
37     struct time a,b;
38     int d;
39
40     a.hour= 8; a.min=26; a.sec=32; /* 8時26分32秒 */
41     b.hour=19; b.min=20; b.sec=15; /* 19時20分15秒 */
42     d=byou(b)-byou(a);
43     printtime (henkan(d));
44 }
```

実行結果

10 時間 53 分 43 秒

問 分数の足し算を行い結果も分数で表すプログラムを作れ。