

## 構造体について

構造体とは、異なる型(int, float, char 等)をまとめて扱うものです。

例えば、以下のような原子に関するデータを扱う時、

char name	char group	float atomicmass
Hydrogen	non-metal	1.00797
Helium	non-metal	4.0026
Lithium	metal	6.939
Berylium	metal	9.0122
Boron	semi-metal-conductor	10.811

これらの値を別々に取り扱うのは好ましくない。この場合には `int main(void)` の前に、

```
struct atoms{
    char name[64];
    char group[64];
    float atomicmass;
};
```

と構造体の内容を宣言し、`main` 文の中で

```
struct atoms c[128];
```

と定義して使う。

この例で、例えば `i` 番目の原子の名前が知りたかったら、`c[i].name` として問い合わせればよいし、原子量が知りたかったら `c[i].atomicmass` とすればよい。

アドレスが知りたければ`&`等を使う。

また、`typedef` という関数を用いて、

```
typedef struct{
    char name[64];
    char group[64];
    float atomicmass;
} atoms;
```

とした場合には、`main` の中で、

```
atoms c[128];
```

と定義して使う。

## 課題

- ① 構造体として,  
`planets(char name[64], int diameter, int distance)`  
を定義し、構造体内のそれぞれの値をキーボード (`scanf` を使う)  
または、ファイル入力 (`fscanf` を使う) から入力できるようにし、  
その入力結果を最後にまとめて表示させなさい。
- ② 上記 `planets` を配列として定義し、複数の惑星についての情報を入力し、  
表示させられるようにしなさい。
- ③ `www` からデータ (`planets.data`) を自分のフォルダにダウンロードし、  
パイプライン (`scanf` を使う) または、ファイル入力 (`fscanf` を使う) により  
10 個のデータを入力しなさい。  
<http://lecture.ecc.u-tokyo.ac.jp/~tohtake/planetsdata.dat>
- ④ 冥王星を除いた新しい構造体配列と作りなさい。  
惑星の名前 `name` を 1 文字のみとし、古い構造体配列からののはじめ一文字としなさい。  
また、太陽からの距離を地球からの距離で正規化した天文単位で表示させなさい。  
これをパイプライン (`printf` を使う) または、ファイル出力 (`fprintf` を使う) により  
ファイル `data2` へ出力しなさい。
- ⑤ ファイル `data2` を入力し、直径順になるように、ソーティングし、表示させなさい。  
ソーティングについては、教科書 (p.60) を参考にすること。
- ⑥ 前回までの配列を用いた課題のうちのいくつかを、動的配列を用いて書き換えよ。

終わった人は最終レポートの作成をしてください

締め切り 2015 年 2 月 19 日 (木) 23:59 厳守

ワードまたは PDF 形式のレポートをメール添付で下記まで提出

[seimitsu.program.2014@gmail.com](mailto:seimitsu.program.2014@gmail.com)

詳細は以下の URL にあります。

<http://lecture.ecc.u-tokyo.ac.jp/~tohtake/FinalReport.pdf>