

## 構造体

構造体とは、C 言語において配列と同じく複数のデータを同時に扱うときに使うものです。配列と違うのは異なる型の変数をまとめることができることです。

### ◆ 使い方

(構造体を宣言するとき)

```
struct 構造体タグ名{  
データの型メンバ名;  
...;  
};
```

(宣言した構造体を使用する時)

```
struct タグ名変数 1, 変数 2;
```

構造体の型自身の名前を構造体タグと呼び、その構成要素をメンバと呼びます。構造体を宣言するときは最後の}のあとの「;」を忘れないようにしてください。

main 文の前に構造体を宣言しておくことで、main 文の中で他の型の変数 (int, double 等) を宣言するときと同じように使うことができます。

### 例

```
#include <stdio.h>  
struct Student{  
int height;  
double bmi;  
};  
int main(void){  
struct Student a, b;  
a.height = 150; a.bmi = 22.5;  
b.height = 175; b.bmi = 19.2;  
printf("Aさん %dセンチメートル、BMI : %f\n", a.height, a.bmi);  
printf("Bさん %dセンチメートル、BMI : %f\n", b.height, b.bmi);  
return 0;  
}
```

この例では、構造体タグ名 : `Student`, メンバ : `height`, `bmi`, 構造体変数名 : `a`, `b` で、二人の生徒の身長と BMI のデータをまとめています。

構造体を用いることで、`int`で扱える範囲の数値と、`double`で扱える数値を同時に扱うことができます。

宣言後にメンバにアクセスする時は、`"a.height"`のように《 変数名.メンバ名 》というようにドット (.) を用いて表します。

また、メンバや変数を配列を用いることもできます。この時にはメンバや変数を指定するときに

```
a[1].height, b.weight[2], c[3].bmi[4]
```

のように[]を用いて表します。

#### ◆ 練習問題

---

1. A さん・B さん・C さんの出席番号と 100m 走のタイムを入力し、3 人の平均タイムを出力するプログラムを作成してください。
2. 1~3 日目の日付 (○日)・製品 A の売り上げ・製品 B の売り上げを入力し、最も売り上げの多かった日の日付と売り上げの合計を出力するプログラムを作成してください。