

# 第7回 C言語勉強会

構造体

佐藤 翼

# 本日の勉強会の流れ

---

**1. 構造体って？**

**2. 構造体の使い方**

**3. 練習問題**

**4. おまけ問題**



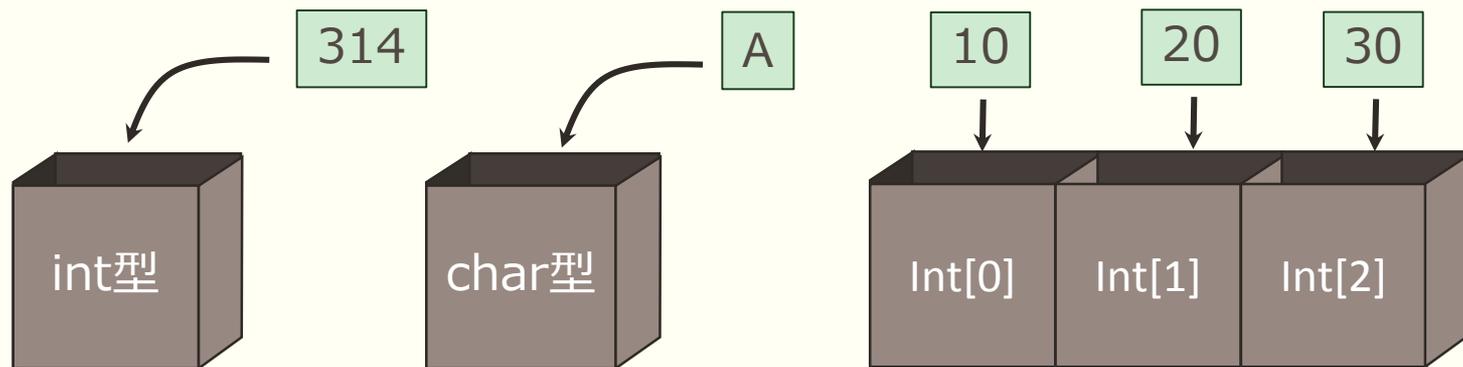
---

# 1. 構造体って？

---

# 構造体の役割

- 今まで色々な変数型や配列を学んできた



- 構造体を使えばこれらをまとめて管理できる

例1) 成績の一括管理

名前	番号	数学	国語	英語
太郎	1	60	50	40
次郎	2	50	40	70
三郎	3	70	40	50

例2) RPGのステータス

キャラ名	レベル	体力
ベル	1	100
コボルト	1	20
スライム	1	10

# 構造体の書き方

例) RPGのステータス

```
struct parameter{  
    char name[20];  
    int lv;  
};
```

構造体の宣言

struct+タグ名

“ ; ”を書き忘れやすいので注意！！

構造体のメンバ  
変数や配列

構造体1

キャラ名  
ベル・クラネル  
Lv  
1

構造体2

キャラ名  
コボルト  
Lv  
1

構造体3

キャラ名  
スライム  
Lv  
1

このように複数の変数を一括で管理できる

# 構造体の書き方

```
#include <string.h>

int main(){
    struct parameter bel;
    strcpy(bel.name, "ベル・クラネル");
    bel.lv = 1;

    printf("%4s : %3s¥n", "Name", bel.name);
    printf("%4s : %3d¥n", "Lv", bel.lv);

    return 0;
}
```

**strcpyを使うために必要**

構造体の変数を宣言  
struct+タグ名+変数名

**文字列をコピー**

構造体への参照  
構造体変数名.メンバ変数名

**実行結果**

```
$ ./pra1
Name: ベル・クラネル
Lv: 1
```

構造体の変数の宣言と値の代入は一緒にできる

```
struct parameter bel = { "ベル・クラネル", 1};
```

**ただし、順番通りに  
代入しなくてははいけない**

# typedefを使ってみよう

---

- 構造体はstruct+タグ名で宣言するので  
 型名(intとか)のように扱っている思えばいい
- これをより分かりやすくするために  
 “typedef”で名前を変えて試してみる

```
typedef int Ten  
Ten number;
```

typedefを使うことで  
“Ten”は“int”と同じ意味で使える

```
typedef struct parameter par ;  
par bel;
```

こうすることで  
“struct parameter”と  
“par”が同じ意味になる



---

## 2. 構造体の使い方

---

# 構造体の入れ子構造

- 構造体のメンバに構造体を作ることにもできる

```
typedef struct {  
    int hp;  
    int str;  
}sta;
```

省略した書き方

```
typedef struct {  
    char name[20];  
    int lv;  
    sta abi;  
}par;
```

構造体のメンバ

```
par bel;  
strcpy(bel.name, "ベル");  
bel.lv = 1;  
bel.abi.hp = 100;  
bel.abi.str = 10;
```

参照の時には  
名前を.でつなげる

```
par bel={"ベル",1,100,10};
```

宣言と一緒に代入する時は  
順番を間違えないようにする

# 構造体の配列

- 構造体を配列として扱うこともできる

```
par enemy[3];
```

配列の宣言と同じようにできる

```
par enemy[] = {  
    { "コボルト", 1, 15, 2 },  
    { "ゴブリン", 1, 20, 3 },  
    { "スライム", 1, 10, 1 },  
};
```

宣言と一緒に代入が可能  
同じく順番には気をつける

参照も同じように

```
printf("%4s : %3s¥n", "Name", enemy[0].name);
```



---

# 3. 練習問題

---

# RPGのステータスを書いてみる

---

- RPGの主人公と敵キャラのステータスを構造体を使って書いて、表示してみる

キャラ名

ベル

Lv

1

HP

10

STR

100

キャラ名

コボルト

Lv

1

HP

15

STR

2

キャラ名

ゴブリン

Lv

1

HP

20

STR

3

キャラ名

スライム

Lv

1

HP

10

STR

1

# 解答

```
#include <stdio.h>

typedef struct {
    int hp;
    int str;
} sta;

typedef struct {
    char name[20];
    int lv;
    sta abi;
} par;

int main(){
    par character[4] = {
        { "ベル", 1, 100, 10 },
        { "コボルト", 1, 15, 2 },
        { "ゴブリン", 1, 20, 3 },
        { "スライム", 1, 10, 1 },
    };
};
```

```
int i;
for (i = 0; i < 4; i++){
    printf("%4s:%3s¥n",
"Name", character[i].name);
    printf("%4s : %3d¥n",
"Lv", character[i].lv);
    printf("%4s : %3d¥n",
"HP", character[i].abi.hp);
    printf("%4s : %3d¥n",
"STR", character[i].abi.str);
}

return 0;
}
```



---

---

## 4. おまけ問題

# RPGみたいに戦闘させてみる

---

- 先ほど作ったステータスを基に自キャラと敵キャラを戦わせてみる

```
$ ./pra5
Name: ペル
  LV:  1
  HP: 100
  STR: 10
1:コボルト  2:ゴブリン  3:スライム  1
ペルの攻撃!コボルトは10のダメージを受けた!
コボルトの攻撃!ペルは2のダメージを受けた!
ゴブリンの攻撃!ペルは3のダメージを受けた!
スライムの攻撃!ペルは1のダメージを受けた!
Name: ペル
  LV:  1
  HP:  94
  STR: 10
1:コボルト  2:ゴブリン  3:スライム |
```

# 解答

- 一緒に掲載している"bell.c"ファイルを参照

第7回C言語勉強会資料(2015/05/01)

講師 : 佐藤翼

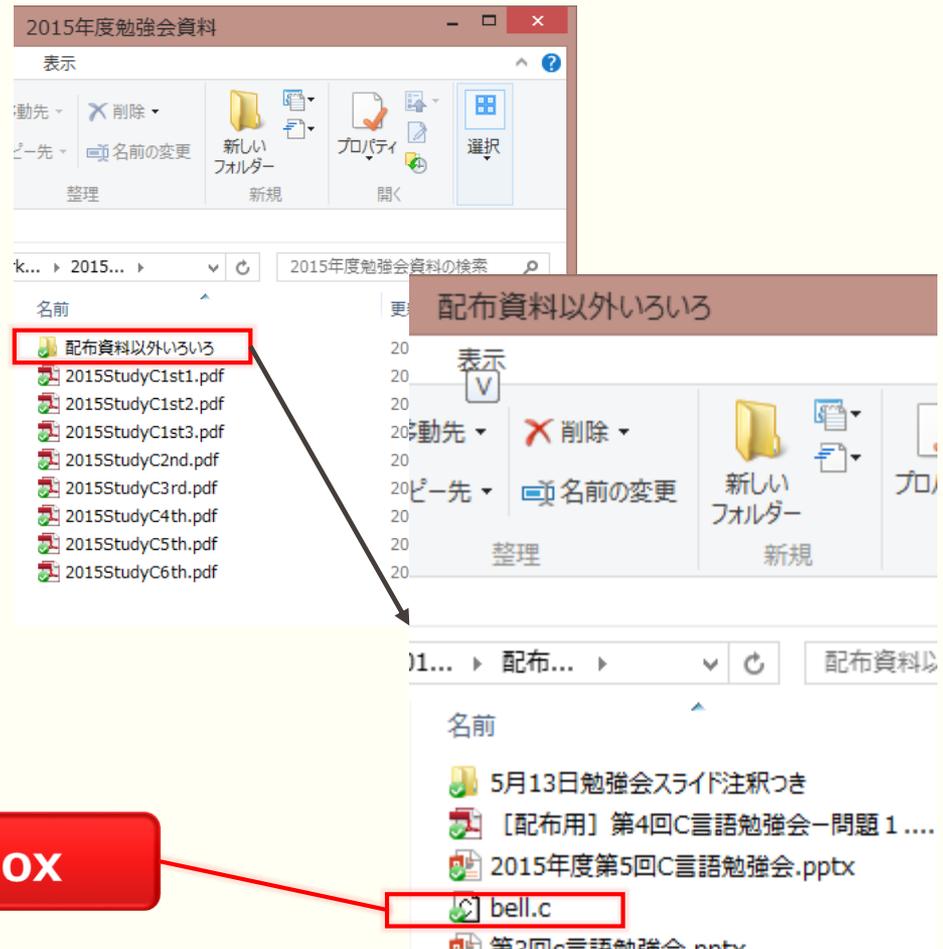
内容 : 構造体

ダウンロード (923KB)

内容 : bell.cファイル

ダウンロード (1.46KB)

HP勉強会資料



DropBox