

# C言語講座

第六回

# 構造体

---

## ▶ 構造体とは？

- ▶ 構造体はint型はchar型などの宣言を枠にまとめて宣言できるようにするもの。
- ▶ 身長や体重などの一つの事柄に関連したデータをまとめて扱うときに便利。
- ▶ 変数宣言に特化した関数みたいなもの。



# 構造体 2

---

## ▶ 枠の定義の仕方

```
struct [構造体名]{  
    [データ型] [メンバー名];  
    (int hogehoge;などの宣言)  
};
```

## ▶ 使うとき

```
struct [構造体名] [変数名] = {初期化};
```

## ▶ 要素の参照

- ▶ 変数名.メンバー名で要素の参照ができる。



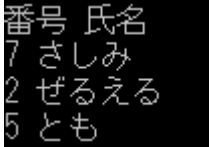
# 例

---

```
struct kouzoutai{
    long no;        /* 番号*/
    char name[20]; /* 氏名*/
};

int main( void )
{
    struct kouzoutai tagu[20]= {
        { 7, "さしみ"},
        { 2, "ぜるえる"},
        { 5, "とも"},
        { 0, ""},
    };
    int i;
    printf( "番号 氏名\n" );
    for ( i=0; tagu[i].no != 0; i++ ) {
        printf( "%d %s\n", tagu[i].no, tagu[i].name);
    }
    return 0;
}
```

## 結果



```
番号 氏名
7 さしみ
2 ぜるえる
5 とも
```



# 文字列

---

## ▶ 文字列

- ▶ 文字列は”ABC”のように”で文字を囲ったもの。
- ▶ 文字列の格納されている変数には末尾に¥0というナル文字が入っている。
- ▶ それによって実際は”ABC”と変数に入れると4文字分の領域をとっている。
- ▶ 文字列は配列で扱う。



# 文字列の格納

---

## ▶ 文字列の宣言

char 配列名[要素数] = 初期化;

□ (例)

char str[] = {'A','B','C','\0'};

または

char str[] = "ABC"; //{"ABC"}でも同じ

▶ 文字列はchar型で宣言。

▶ scanfでキーボードから取得させるときは%sを使う

▶ (例) scanf("%s", str);

▶ 配列の大きさを指定した場合、余った領域にはナル文字¥0が格納される。

▶ 一文字取得のときにはgetchar関数でもできる。

▶ str[i] = getchar();

---

# 文字列で使うと便利な関数

---

## ▶ strcpy関数

- ▶ 別の文字型配列にコピーする。
- ▶ strcpy( コピー先 , コピー元 )
  - ▶ (例) strcpy(str1, str2); //str1にstr2をコピーする

## ▶ strcmp関数

- ▶ 文字列を比較する。
- ▶ 同じなら0、異なるなら-1か1を返す。
- ▶ strcmp(比較文字列1 , 比較文字列2);
  - ▶ (例) strcmp(str1, str2)



# 文字列で使うと便利な関数2

---

## ▶ strlen関数

- ▶ 文字列の長さを返す。(ナル文字までを調べる)
- ▶ strlen(文字列)
  - ▶ (例) strlen(str1)

## ▶ strcat関数

- ▶ 文字列を連結する。
- ▶ strcat(繋げられる文字列, 繋げる文字列)
  - ▶ (例) strcat(str1, str2);

## ▶ strncat

- ▶ 文字列を元の文字列の後ろにn文字連結させる。
- ▶ strncat(繋げられる文字列, 繋げる文字列, 繋げる文字の数)
  - ▶ (例) strncat(str1, str2, n)

- ▶ 注意: これらの関数を使うときは#include<string.h>を宣言する。



# 練習

---

- ▶ 1,キーボードから英数字を入力し、含まれているアルファベットの個数を数える。
- ▶ 2,50文字になるまで入力された文字列を後ろに連結させていき、50文字になったら表示させる。



# 1, 答えの例

---

```
int main()
{
    char str[20] = "0";
    scanf("%s",str);

    int i;
    int count = 0;

    for(i=0;i<strlen(str);i++){
        if('0' <= str[i] && '9' >= str[i])
            printf("数字¥n");

        else{
            printf("アルファベット¥n");
            count++;
        }
    }
    printf("文字列に含まれるアルファベットの個数は%d個¥n",count);

    return 0;
}
```



## 2, 答えの例

---

```
int main()
{
    char str_a[100] = ""; //NULL文字で初期化
    char str_b[51] = ""; //50文字+ナル文字なので51

    while(strlen(str_b) < 50) {
        printf( "文字列を入力してください" );
        scanf( "%s",str_a);
        strcat( str_b, str_a, 50 - strlen(str_b));
    }
    printf( "長さ= %d¥n%s", strlen( str_b ),str_b);

    return 0;
}
```

