

C 言語 II 演習(2-2) ~文字列(2)~ 教科書 9章 解答例

*以下はプログラム例であり、全く同じである必要はありません。プログラム作成の参考として確認してください。

問題 1.

```
// 演習(2-2) 文字列 問題1
#include<stdio.h>

int main(void) {
    char str[5]="KAIT" ;
    // 4文字入れるには5個の箱が必要です

    printf("文字列は%#n %s %#n です#\n", str);

    return 0;
}
```

問題 2.

```
// 演習(2-2) 文字列 問題2
問題1の解答例について次を修正する。

#include<stdio.h>

int main(void) {
    char str[21];
    // 20文字入れるには21個の箱が必要です
    printf("文字列を入力してください#\n");
    gets(str); // scanf はスペース含めないのがダメ
    printf("入力された文字列は %s です。#\n", str);

    return 0;
}
```

問題 3.

```
// 演習(2-2) 文字列 問題3
#include<stdio.h>

int main(void) {
    char str[30];
    int i;

    printf("29文字以内の文字列を入力してください。#\n>");
    scanf("%s", str); // gets(str);でも良い。
    printf("入力された文字列は次です。#\n%s#\n", str);

    for(i=0; ; i++){
        if(str[i]=='a') {
            printf("入力文字列の中にaがありました。#\n");
        }
        if(str[i]=='\0') { break; }
    }

    return 0;
}

/* 参考: 上記ソースの for 文 6行は次に置き換えても良い。
for(i=0; str[i]!='\0'; i++){
    if(str[i]=='a') {
        printf("入力文字列の中にaがありました。#\n");
    }
}
*/
```

【getchar() の活用】

次の 1) や 2) を含むプログラムでは、

- 1) 一時停止用の `getchar()` ;
- 2) `char` 型 1 文字読込用の `scanf("%c", &ch)` ;

1) や 2) の記述があるにも関わらず、この箇所が正常に実行されない (処理が飛ばされて無視されるような現象が生じる) 場合がある。

このような場合、プログラム内の「正常に実行されない 1) や 2) より手前の部分」に存在する 1 つまたは複数の `scanf()` 文の直後に、次を追記すると良い。

「入力に続く改行文字を読み飛ばす」ための `getchar()` ;

(以上)