

2022 年度
プログラミング第 1 同演習

第 9 回

慶應義塾大学 理工学部 情報工学科

講義担当：河野健二

演習担当：杉浦裕太

【課題の提出方法】

提出時期：

- 次の授業日の午前 9:00
- 遅れた場合は減点されます
- 遅れる場合でも、さらにその次の授業日の 9:00 までには提出をしてください（提出フォームが自動的に閉じます）

提出先：

- CANVAS 経由

提出方法：

- 「xxxxxx.c」 ファイルを一つずつ CANVAS にアップロードして提出
- 課題を再提出したい場合は、変更した課題だけではなくすべての課題を再アップロードするようにしてください。（以前にアップロードしたものは消えてしまう可能性があります）

ファイル名：

- 課題のスライドのタイトルと同じ（採点時に重要なので必ず同じにしてください）。ただし課題を再アップロードする際はファイル名が変更されることもあり、これに関してはそのままで大丈夫です

終了対応：

- 課題が終わったら各自退出していただいてかまいません
- 授業終了の時間がきて課題が終わっていなくても帰宅していただいてかまいません
- 残りの課題は ITC パソコンか各自のパソコンで行ってください

ITC Linux パソコン以外で作業する際の注意点：

- 自分の環境で発生した問題は自己責任でお願いします
- 提出するプログラムコードは Linux 環境でコンパイルできることを確認すること
- 採点時に、Linux 環境でコンパイルできることを確認するため、その時点でコンパイルが失敗してしまった提出物の点数は自動的に 0 点になってしまいます

【必須課題】

09-a-01.c

【課題】

ascii コードを入力すると対応する文字が出力されるプログラムを作成してください。

【実行例：入力赤字】

```
123
{
---
100
d
---
43
+
```

【解答例】

09-a-02.c

【課題】

入力された 1 行の文字列が回文かどうか判定するプログラムを作成してください。入力は `getchar()`関数で行うこと。入力文字列は最大でも 20 文字までとする。

【ヒント】

入力は配列などに格納する。

【実行例：入力は赤字】

```
abcdefgfedcba
This str is a palindrome
---
abcabc
This str isn't a palindrome
---
a
This str is a palindrome
```

【解答例】

09-a-03.c

【課題】

float 型配列と配列の要素数を引数にとり、配列の全要素を 0.0 で初期化する関数 void init_array(float [], int)を作成し要素数 10 の float 型配列を初期化してください。

【実行例：入力赤字】

```
f[0]:0.000000
f[1]:0.000000
f[2]:0.000000
f[3]:0.000000
f[4]:0.000000
f[5]:0.000000
f[6]:0.000000
f[7]:0.000000
f[8]:0.000000
f[9]:0.000000
```

【解答例】

09-a-04.c

【課題】

要素数 10 の `char` 型配列に文字を入力し、格納された文字の内何文字が小文字の母音 (`a,i,u,e,o`)かを数える関数 `int counter(char [], int)`を作成し、動作を確認するプログラムを作成してください。関数の第二引数は配列の要素数。

【実行例：入力は赤字】

```
input 10 characters :AsDertYUio  
count : 3
```

【解答例】

09-a-05.c

【課題】

実行例のように二次元配列を表示する関数 `void disp_array(int[][4])` を作成してください。

【ヒント】

main 関数は以下のように記述すること。

```
int main()
{
    int array[3][4] = { {11,12,13,14},
                        {15,16,17,18},
                        {19,20,21,22} };

    disp_array(array);

    return 0;
}
```

【実行例：入力は赤字】

```
11 12 13 14
15 16 17 18
19 20 21 22
```

【解答例】

09-a-06.c

【課題】

2 つの整数を入力し，その四則演算，ビットごとの AND, OR, XOR, 左シフト, 右シフトを 求めるプログラムを作成しなさい. 加減乗除は 10 進数, ビットごとの AND, OR, XOR, 左シフト, 右シフトは 16 進数で表示しなさい.

【ヒント】

16 進数表示は `printf("%x")`

【実行例：入力は赤字】

```
Input a: 10
Input b: 3
10 + 3 = 13
10 - 3 = 7
10 * 3 = 30
10 / 3 = 3
0xa & 0x3 = 0x2
0xa | 0x3 = 0xb
0xa ^ 0x3 = 0x9
0xa << 0x3 = 0x50
0xa >> 0x3 = 0x1
```

【解答例】

【標準課題】

09-b-01.c

【課題】

3x3 の行列を入力し、加算を行うプログラムを作成せよ。以下の関数を定義して使用すること。

void get_arr3x3(int a[][3]) : 標準入力から数字を読み込み、a00~a22 の順で数字を格納する。

void sum_arr3x(エックス)3(int a[][3], int b[][3], int c[][3]) : a + b の結果を c に格納する。

void print_arr3x3(int a[][3]) : 3x3 行列を標準出力に表示する。

【実行例：入力値は赤字】

1 2 3 4 5 6 7 8 9[Enter]

a

1 2 3

4 5 6

7 8 9

9 8 7 6 5 4 3[Enter]

2 1[Enter]

b

9 8 7

6 5 4

3 2 1

a + b

10 10 10

10 10 10

10 10 10

【解答例】

09-b-02.c

【課題】

複数の数字、アルファベットを入力とし、その中から 0 が何番目に存在するかを表示するプログラムを作成してください。getchar 関数を用いること。関数 void find_zero(char *a, int n)を定義して使用すること。

【実行例：入力赤字】

abcd0123ef01gh23[Enter]

Zero in 4 10(注：一番先頭が 0 番目)

【解答例】

09-b-03.c

【課題】

EOF が入力されるまで文字の入力を受け付け、**Enter** キーを押す度に入力された文字の `ascii` コードを出力するプログラムを作成してください。

【ヒント】

`getchar()`関数で読み込まれる改行に対処すること。

【実行例：入力は赤字】

```
a c[Enter]  
97  
32  
99  
12[Enter]  
49  
50  
^d
```

【解答例】

09-b-04.c

【課題】

入力したアルファベット文字を `ascii` コード順に出力するプログラムを作成してください。何らかの数値が入力されるまで入力が行われるものとする。`getchar()`関数を使用すること。

【ヒント】

入力文字列の長さが不定である以上、文字データの格納はリストを使うことが望ましい。

【実行例：入力は赤字】

```
abfGageGaheifhalenbcu
afuhneixlDbfu
hflfi0
DGGaaaaabbbceeeefffffghhhhiiillnnuuux
```

【解答例】

09-b-05.c

【課題】

標準入力から 1 文字入力し，2 進数に変換するプログラムを作成しなさい．ただし，1 文字は 1byte(8bit) で表されると仮定してよい．

【実行例：入力は赤字】

```
e
01100101
---
Z
01011010
---
@
01000000
```

【解答例】

--

09-b-06.c

【課題】

入力された符号なし整数 (`unsigned int` 型) のビット列において、下位ビットから数えて何番目に最初の 1 があるかを調べるプログラムを作成しなさい。実行例の様に、一行目で 16 進数の整数の入力を促し、結果を `bit index` の値として出力させなさい。なお、`unsigned int` 型のビット長は 32 ビットであると仮定してよい。
また、ビットは最下位ビットを 0 とし、最上位ビットを 31 とする。ただし、1 が 1 つもない場合には 32 を出力しなさい。

【ヒント】

16 進数の読み取りは `scanf("%x")`

【実行例：入力は赤字】

```
word = 0x1
bit index = 0
---
word = 0xffffffff
bit index = 0
---
word = 0xf0
bit index = 4
---
word = 0x80000000
bit index = 31
---
word = 0x0
bit index = 32
```

【解答例】

--

【ボーナス課題】

09-c-01.c

【課題】

`getchar` 関数を用いて文字列を読み込み、区切り(スペース、改行、ピリオド)ごとに先頭文字を大文字に、それ以外を小文字にするプログラムを作成せよ。必要な関数は各自定義して使用してよい。

【実行例：入力は赤字】

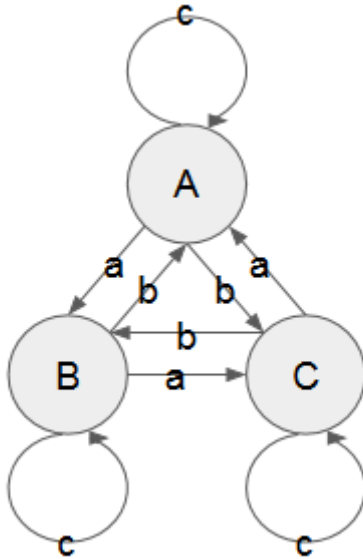
```
heLLo WoRLD.i'm student. Programming TANOSHIII!  
Hello World.I'm Student. Programming Tanoshiii!  
c is programming language.  
C Is Programming Language.  
^d
```

【解答例】

09-c-02.c

【課題】

図のような決定有限性オートマトンへの入力(a,b,c)による状態遷移を出力するプログラムを作成してください。



【ヒント】

初期状態(place[0])は A とする。
getchar 関数および switch 文を用いること。
想定外の入力が行われたら遷移を終了すること。

【実行例：入力は赤字】

abcbcb[Enter]

place[0] : A
place[1] : B
place[2] : A
place[3] : A
place[4] : C
place[5] : C

bcd[Enter]

place[0] : A
place[1] : C
place[2] : C

【解答例】

【チャレンジ課題】

09-d-01.c

【課題】

`getchar` 関数を用いて逆ポーランド記法で書かれた式を計算せよ。必要な関数は各自定義して使用してよい。

式は正しく入力されるものと仮定してよい。除算は入力されないものとする。`stdio.h` 以外のライブラリは使用してはならない。

逆ポーランド記法の例

$2 + 3 \rightarrow 2\ 3\ +$

$5 + 20 * 3 \rightarrow 5\ 20\ 3\ *\ +$

$(1 + 15) * (23 - 3) \rightarrow 1\ 15\ +\ 23\ 3\ -\ *$

【ヒント】

計算結果を保存するスタック、読み込んだ数字の列を数値に変換するためのスタックを用意するとよい

`push`, `pop` などの基本操作は関数を用いるとよい。

【実行例：入力は赤字】

2 3 + [Enter]

Result: 5

5 20 3 * + [Enter]

Result: 65

1 15 + 23 3 - * [Enter]

Result: 320

【解答例】