

Homework #5

C Programming

Mission



- Solve problem 1~3
- Submission
 - Submit a zip file containing hw5_1.c, hw5_2.c, and hw5_3.c on HISNet.
 - Source files hw5_*.c should contain the followings:
 - Algorithm in pseudo code as comments preceded by “//”.
 - See the next page.
 - C program code that implements the pseudo code
 - cf. Upload only source files (.c files)
 - Search for *.c files under the project directory and copy them to hw5_1.c, hw5_2.c, or hw5_2.c.
 - The code should be properly indented.
- Due date: PM 11:00, Nov. 1st

Code Example (add.c)

```
#include <stdio.h>

int main()
{
    int a = 0, b = 0;
    int sum = 0;

    // read two integer numbers
    printf("Input two numbers: ");
    scanf("%d %d", &a, &b);

    // add the two numbers
    sum = a + b;

    // print the result
    printf("%d + %d = %d\n", a, b, sum);

    return 0;
}
```

Indentation is Crucial for Readability



■ Properly indented code

```
int main()
{
    int height = 0;
    int i = 0, j = 0;

    do {
        printf("Input the height of triangle: ");
        scanf("%d", &height);
    } while (height % 2 == 0);

    for(i = 1; i <= height; i++){
        int start = height - i;
        int end = height + i - 1;

        for(j = 0; j < start; j++)
            putchar(' ');
        for(; j < end; j++)
            putchar('*');
        putchar('\n');
    }

    system("PAUSE");

    return 0;
}
```

■ Not indented code

```
int main()
{
    int height = 0;
    int i = 0, j = 0;

    do {
        printf("Input the height of triangle: ");
        scanf("%d", &height);
    } while (height % 2 == 0);

    for(i = 1; i <= height; i++){
    int start = height - i;
    int end = height + i - 1;

    for(j = 0; j < start; j++)
    putchar(' ');
    for(; j < end; j++)
    putchar('*');
    putchar('\n');
    }

    system("PAUSE");

    return 0;
}
```

Rules of Indentation

- Start function header from the first column
- Use tab to represent indentation level.
- The body of functions, blocks, selection (if, switch), or repetition (while, for, do-while) statements should be indented one more level.

Ex) if(x % 2 == 0)

↔ printf("x is an even number.\n");

Ex) do {

↔ printf("Input a positive number:");

↔ scanf("%d", &x);

} while(x <= 0);

Honor Code Guidelines (Korean)

■ “과제”

- 과제는 교과과정의 내용을 소화하여 실질적인 활용 능력을 갖추기 위한 교육활동이다. 학생은 모든 과제를 정직하고 성실하게 수행함으로써 과제에 의도된 지식과 기술을 얻기 위해 최선을 다해야 한다.
- 담당교수가 명시적으로 허락한 경우를 제외하고 다른 사람이 작성하였거나 인터넷 등에서 획득한 과제물, 또는 프로그램 코드의 일부, 또는 전체를 이용하는 것은 부정행위에 해당한다.
- 자신의 과제물을 타인에게 보여주거나 빌려주는 것은 공정한 평가를 방해하고, 해당 학생의 학업 성취를 저해하는 부정행위에 해당한다.
- 팀 과제가 아닌 경우 두 명 이상이 함께 과제를 수행하여 이를 개별적으로 제출하는 것은 부정행위에 해당한다.
- 서로 다른 학생이 제출한 제출물간 유사도가 통상적으로 발생할 수 있는 정도를 크게 넘어서는 경우, 또는 자신이 제출한 과제물에 대하여 구체적인 설명을 하지 못하는 경우에는 부정행위로 의심받거나 판정될 수 있다.

Problem 0a: Recursion



- Compile run the recursion example programs
 - Fibonacci.c
 - GetDigit_recursion.c
- Read the code and their results carefully to understand the recursion.

Problem 0b: File I/O



- Remember all the file I/O techniques exhibited in the following code

```
// copy.c
#include <stdio.h>

int main()
{
    FILE *in = NULL, *out = NULL;
    int c = 0;           // Note that c should be of int type

    // open source and destination files
    in = fopen("src_file.txt", "r");
    out = fopen("dest_file.txt", "w");

    // copy all characters from src_file to dest_file until the fgetc() returns EOF
    while((c = fgetc(in)) != EOF)    // Repeat to read a character from src_file until EOF is read
        fputc(c, out);              // write the character into dest_file

    // close files
    fclose(in);
    fclose(out);

    return 0;
}
```


Problem 1



- Read text lines until the user inputs an empty string (type Enter at the first column). Then, print the numbers of uppercase chars, lowercase chars, digits, and whitespace chars.
 - Use `isspace()` to count whitespace characters.
 - You need to include `ctype.h`.
 - You may use other `isXXX()` functions.

Example)

Input a text. Type a blank line to finish.

Welcome to HGU.

HGU was founded in 1995.

Most students in this class have the student ids of 2018.

I hope everyone achieves a lot.

// the user hit Enter at the first column

The text has 9 uppercase chars, 85 lowercase chars, 8 digits, and 21 whitespace characters.

Problem 1



■ Algorithm

Repeat following actions

Read a text line from the user

// Use gets() to read a text line

If the input is an empty string, break the loop

// Check if the first character is '\0' to detect an empty string.

Count each character types in the text line

Print the results

Problem 2



- Read the name of a text file. Then, print the numbers of uppercase chars, lowercase chars, digits, and whitespace chars.
 - Combine your solution of Problem 1 and the techniques you have learned from copy.c

Ex) A text file “data.txt” has the following contents:

Welcome to HGU.

HGU was founded in 1995.

Most students in this class have the student ids of 2018.

I hope everyone achieves a lot.

Input filename: **data.txt**

The text has **9** uppercase chars, **85** lowercase chars, **8** digits, and **24** whitespace characters.

- Note that # of whitespace characters can be slightly different on Mac or Linux.

Problem 3



- Write a function “`void RotateString_Left(char str[]);`”
 - RotateString_Left() should rotate all character in *str* one step left.

Example)

```
char string[256] = "God bless you!";  
printf("Before: %s\n", string);  
RotateString_Left(string);  
printf("After: %s\n", string);
```

Result)

```
Before: [God bless you!]  
After: [od bless you!G]
```

Problem 3



- Combined with main() on the next page, the output should be as the following example.

Example)

Input a text line: **Nice to see you!**

The length of string "Nice to see you!" is 16.

text = [Nice to see you!]

text = [ice to see you!N]

text = [ce to see you!Ni]

text = [e to see you!Nic]

text = [to see you!Nice]

text = [to see you!Nice]

text = [o see you!Nice t]

text = [see you!Nice to]

text = [see you!Nice to]

text = [ee you!Nice to s]

text = [e you!Nice to se]

text = [you!Nice to see]

text = [you!Nice to see]

text = [ou!Nice to see y]

text = [u!Nice to see yo]

text = [!Nice to see you]

text = [Nice to see you!]

Problem 3



- Main() function

```
#include <stdio.h>
#include <string.h>           // This is for strlen()

void RotateString_Left(char str[]);

int main()
{
    char text[256];
    int len = 0;
    int i = 0;

    printf("Input a text line: ");
    gets(text);

    len = strlen(text);       // Then, len will contain the length of text
    printf("The length of string W\"%sW\" is %d.Wn", text, len);

    printf("text = [%s]Wn", text);
    for(i = 0; i < len; i++){
        RotateString_Left(text);
        printf("text = [%s]Wn", text);
    }

    return 0;
}
```