Homework #3

C Programming

Mission

- Solve problem 1~3
- Submission
  - Submit a zip file containing hw3_1.c, hw3_2.c, and hw3_3.c on HISNet.
  - Source files hw3_*.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 hw3_1.c, hw3_2.c, or hw3_2.c.
    - The code should be properly indented.
- Due date: PM 11:00, Oct. 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

- Star 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 0

- Memorize the three control statements. (Quiz on Oct.1)
  - if-statement

```
if(<condition>){                    // ex) x % 2 == 0, x < y
    <statement1>                    // executes when <condition> is true
} else {                            // 'else' part can be omitted
    <statement2>                    // executes when <condition> is false
}
```

  - for-statement

```
int i = 0;                          // loop variable
for(i = 0; i < N; i++){             // N : # of iterations
    <statement>
}
```

  - while-statement

```
while(<condition>){                 // repeat while <condition> is true
   <statements>
}
```

Problem 1

- Write “number finding” game.

  Ex)

  I selected a number between 1 and 100. Find it by guess.
  Input a number between 1 and 100: 50
  50 is too small.
  Input a number between 1 and 100: 75
  75 is too small.
  Input a number between 1 and 100: 80
  80 is too large.
  Input a number between 1 and 100: 78
  Great! You got it. My number is 78!

Problem 1

- Algorithm
  - Computer selects a number x between 1 and 100. Then print a message to promote the user to input numbers.
    - Use “srand(time(NULL));” and “x = rand();” to generate random numbers
  - Repeat until the user inputs the correct number
    - Read a number $guess$ between 1 and 100
    - If $guess$ is smaller than x, print “$guess$ is too small.”
    - Otherwise, if $guess$ is larger than x, print “$guess$ is too large.”
    - Otherwise, if $guess$ is the same as x, print a congratulation message

Problem 2

- Write an Arithmetic Problem Generator.
  - Generate a random arithmetic expression
  - Read answer from the user
  - Decide whether the answer is correct or not

Ex)

```
1 + 2 = 3
Correct! Your are genius!
10 / 7 = 1
Correct! Your are genius!
10 - 4 = 6
Correct! Your are genius!
9 - 8 = 2                                   // wrong answer
The correct answer is 1. Try again!
6 - 3 = 8                                   // wrong answer
The correct answer is 3. Try again!
4 * 10 = 40
Correct! Your are genius!
5 / 8 = -999                                // quit
Bye!
```

Problem 2

- Algorithm
  - Repeat infinitely
    - Randomly select two integers a and b between 0 and 9
    - Randomly select an operator op among { +, −, *, / }
    - Display the binary expression composed of a, b, and op
    - Compute the correct answer (use if or switch statement)
    - Read the answer from the user.
    - Decide whether the answer is correct or not. Print an appropriate message
    - If the answer is −999, break the loop

Problem 2

- Hints
  - Use “while(1)” to implement an infinite loop.
    - Any non-zero integer represents $true$ in logic expression
  - Use “srand(time(NULL));” and “rand()” to generate random numbers
  - Randomly selecting operator

```
char op = 0;
char operator[5] = "+-*/";

// for a string s, s[i] retrieves  i^th character of s.
op = operator[rand() % 4];
```

  - Compute the correct answer using if-statement or switch-statement

Problem 3

- Draw a reversed triangle.
  - Read the height of the triangle from the user

Ex)

Input the height of triangle: 10

```
**********
*********
********
*******
******
*****
****
***
**
*
```

- Hint:
  - Learn required techniques from the next page

Problem 3

- Hint: Drawing triangle (NOT reversed)

```
#include <stdio.h>

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

    printf("Input the height of triangle: ");
    scanf("%d", &height);

    for(i = 1; i <= height; i++){
        for(j = 1; j <= i; j++)
            printf("*");
        printf("₩n");
    }

    return 0;
}
```

Ex)

```
Input the height of triangle: 10
*
**
***
****
*****
******
*******
********
*********
**********
```