C Programming I
2020 Fall
Homework 03

Instructor: Po-Wen Chi

Due: 2020.11.3 PM 11:59

Policies:

- Zero tolerance for late submission.
- You need to prepare a README file about how to make and run your program. Moreover, you need to provide your name and your student ID in the README file.
- For the writing assignment, I only accept pdf. MS. doc/docx format is not acceptable. Moreover, please use Chinese instead of English.
- Do not forget your Makefile. For your convenience, each assignment needs only one Makefile.
- The executable programs should be hw0301, hw0302 ....

1 Tornado (20 pts)

Please develop a program to make the user input two integers as the width m and the height n. Your program should list all integers from 1 to $m \times n$ as follows. Start from the top left corner and list these numbers inward in a clockwise direction. Note that alignment is an issue in this problem.

```
$ ./hw0301
Please enter the width : 5
Please enter the height: 3
 1  2  3  4  5
12 13 14 15  6
11 10  9  8  7
```

You should give an error message and terminate your program when receiving an invalid input.

2 Swap Digits (20 pts)

Please write a C program to input a natural number from user and swap first and last digit of the given number. In this problem, I guarantee that the input number can be stored in a 32-bits memory block.

```
$ ./hw0302
Please enter a natural number : 12345
52341
```

You should give an error message and terminate your program when receiving an invalid input.

3 Finite State Machine (20 pts)

What is Finite State Machine? Well ... there is an important class called **Automata** taught by Professor Hou. I suggest you to take this course. So I will not introduce what it is but simply describe how it works. Figure 1 is a finite state machine example. The circle implies the state. Initially you are in the start state. When receiving a number, you will move to the next state according to the red number indication. The word **all** means you will move to this state with all input numbers. The word **others** means you will move to this state with all input numbers except the number listed to other states.

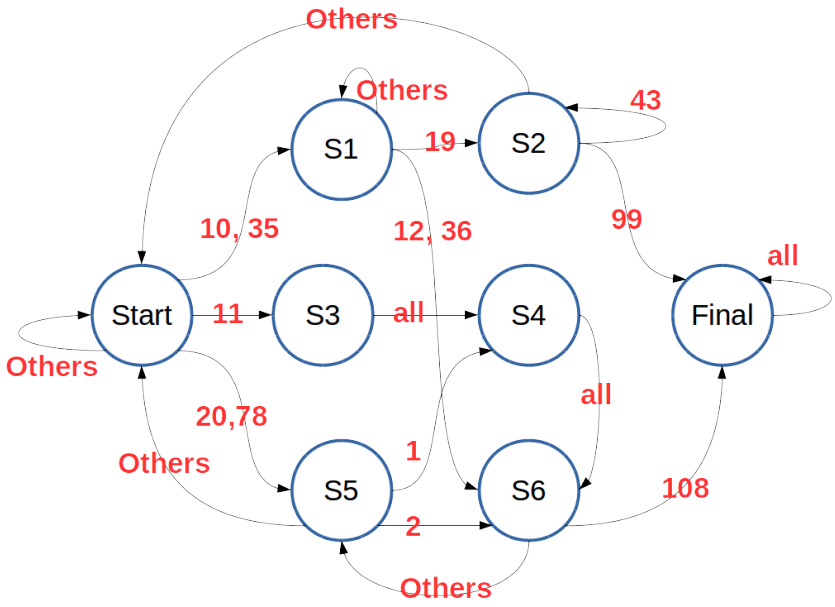

Figure 1: Deterministic Finite Automata.

Now you need to develop a program for a user to input integers until the final state.

```
$ ./hw0303
Start
Please enter an integer: 35
S1
```

```
Please enter an integer: 19
S2
Please enter an integer: 43
S2
Please enter an integer: 99
Final
```

4 Definite Integral (20 pts)

In mathematics, an integral assigns numbers to functions in a way that can describe displacement, area, volume, and other concepts that arise by combining infinitesimal data. I believe that you know how to calculate the definite integral of a polynomial, right? If not, you can ask TA for help.

When studying integral, I believe that your teacher has shown you **Riemann integral**, an integral definition. Given a polynomial $f(x)$, you can calculate the Riemann sum of $f(x)$ in the interval $[a, b]$ as follows.

$$\sum_{i=0}^{n-1} f(a + \frac{b-a}{n} \times i) \times \frac{b-a}{n}.$$

When $n \to \infty$, the Riemann sum is close to the Riemann integral. Now I want you to develop a program to simulate this process. For your simplicity, the polynomial in this problem is a quadratic polynomial. You need to calculate the Riemann sums from $n = 2$ to $n = 65536$.

```
$ ./hw0304
Please enter a quadratic polynomial (ax^2+bx+c): 1,0,0
The polynomial is x^2
Please enter the interval [s,t]: 0,1
The integral: 0.333333
The Riemann sum of n=2: 0.125
The Riemann sum of n=4:
The Riemann sum of n=8:
...
The Riemann sum of n=65536:
```

Please use **double** in this program. The precision is not a concern in this problem. You need also draw a figure to show if the Riemann sum is gradually close to the definite integral. If no, please explain the reason.

You should give an error message and terminate your program when receiving an invalid input. For example, t must be larger than s.

5 Systematic Investment Plan (20 pts)

A systematic investment plan (SIP) is a plan where investors make regular, equal payments into a mutual fund, trading account, or retirement account.

I will give you an example first. Please see the following DM as an example.

https://www.fubon.com/life/cms/C3CDF70FA2D245079A22E0D7F5CD30BE/2019-01/20200828132803610157617 7.pdf

I have to say, this is just an example, not advertisement. Please read the table in page 2. I will show you how to calculate the value in the first year.

The condition is as follows:

- Annually payment: 40000
- Insurance amount: 1000000
- Insurance fee:
  - 1st year: 60%
  - 2nd year: 30%
  - 3rd year: 30%
  - 4th year: 15%
  - 5th year: 15%
  - 6th and after: 0
- Insurance fee off: 1%
- Insurance cost: depends on sex and age which is 2256 in the first year of this example.
- Insurance processing fee: 100 monthly.
- Annual return on investment rate: 5%.

So in the first year, the actual value of your account is

$$40000 \times (1 - 60\% + 1\%) - 2256 - 1200 = 12944.$$

After **monthly compound interest** in one year, you can get 13674. And if you die or become disabled, you, or your children, can get 1013674. Now I want you to develop a SIP calculator. You should calculate the account value from the start year to 100 years old. For your simplicity, the insurance cost is defined as $100 \times (\text{age} - 20) + 2000$ and 2000 if the age is less than 20.

```
$ ./hw0305
SIP Plan
Annually payment: 40000
Insurance fee in the first five years (0 is assumed afterwards):
60,30,30,15,15
Insurance fee off: 1
Monthly insurance processing fee: 100
Age: 40
How many years of payment: 15
Expected annual return on investment rate: 5
------------------------------
Your Payment and Account Value Table
```

```
40: 40000, 13671
41: 80000, 40453
...
100:
```

Note that 13671 and 40453 are not correct because we simplify the the insurance cost and I am lazy to calculate the correct value. The year of payment is that you only pay 40000 in the first 15 years.

You should give an error message and terminate your program when receiving an invalid input. Again, precision is not an issue this problem. When the account value is less than 0, this investment plan will stop.

6 Bonus: What is the problem? (10 pts)

Please read the following code. How many "Hello Kitty" will be printed? I think the answer should be four, right?

```
#include <stdio.h>


int main()
{
    for( double f = 0.0; f <= 0.3; f += 0.1 )
    {
        printf( "Hello Kitty.\n" );
    }

    return 0;
}
```

However, in my computer, the output is as follows.

```
~ $ ./a.out
Hello Kitty.
Hello Kitty.
Hello Kitty.
```

What is the problem? Please write down your explanation with proofs that support your opinion.