CIS2107_Lab03: "ATM"

Points: **100** points

Mission:

- Write a C Program for an ATM machine by implementing functions.
- To protect your program from suspicious activities.
- Attackers, thieves, invaders, and users acting smart will do every possible act to steal data and money, and ruin your program.

Objectives:

- To use welcoming and closing messages.
- When it comes to outputs, be creative. Make your application friendly and easy to use.
  - Implement **\n**, **\t**, **" ", and $** sign as needed.
- To use numerical calculations and Mathematical operations **"+, -, *, /, %"** to solve a real-life challenge.
- To catch invalid numerical inputs (Negative numbers and Zeroes)
- To use `exit(0)` to end the program. To do so, add `#include <stdlib.h>` to your program.

Rules:

Write a C Program for an ATM machine.

The types of ATM transaction are:

1. *Balance*.
2. *Cash withdrawal*.
3. *Cash deposition*.
4. *Quit*.

All ATM transactions (menu options) need to be implemented in terms of functions.
You will call those functions based on the menu option.

- Initially, Nana has one checking account with a balance of $5000 with PIN number of 3014.
- Money amount validation:
  - Catch invalid numerical inputs (Negative numbers and Zeroes).
  - Users can enter numbers only.
  - Remember any ATM has numpad only. This numpad has neither letters nor special characters.
  - Don't worry about entering other inputs rather than numbers for now.
  - I mean don't worry what would happen if the user enters a string, a char, or a special characters instead of a number. We will discuss this later in another chapter.
  - In short, if the Nana enters wrong amount, ask her to enter amount again, after 3 unsuccessful attempts, display an error message and terminate the program.
- PIN validation:
  - Nana has 3 attempts to enter PIN. After that the program logs them off, and end the program.
- Once Nana chooses to finish using the ATM, print out Thank You message and tell her the number of transactions she has made.
- All ATM transactions (menu options) need to be implemented in terms of functions.
  - You will call those functions based on the menu option.
- All ATM transactions have receipts, but it is up to the user to choose.
  - Use 1 for yes, and 2 for No.
  - The receipt is virtual. No need to display the receipt on screen.

- *Balance*:
    - Nothing special about it. Basically it prints out the balance.

- *Cash withdrawal*:
    - The limit is $1000 a day.
    - For every withdrawal transaction, there is a receipt.
    - Nana enter the amount to withdraw in multiples of 20s. (i.e.; 20, 40, 60, 80, and so on).
    - ATM machine cannot output 1s, 5s, or 10s paper bills only, and of course No coins.

- *Cash deposition*:
    - The limit is $10000 a day.
    - For every withdrawal transaction, there is a receipt.
    - Nana enter the amount to deposit.
    - ATM machine accepts any paper bills including 1s, 5s, and 10s, but of course No coins.

- *Quit*:
    - From its name, you will the number of transactions and Thank You message program.
    - Finally, end the program.

Submission:

- Be sure to document your code (add comments on top of your C file). In the comments add your name, date, course, homework number, and statement of problem.
- Once you are done, upload one C program called `atm` through Canvas.