Mahidol University

ITCS 201 – Fundamentals of Programming
Week 13: Lab Assignments

Name: ____ **ID:** ________________

Due: today or in a lab session next week

Instructions:

- Marking lab assignments will be done in the lab
- **Compile** and **Run** your program
- **Show** and **Explain** the output and your code to the lecturer or the lab assistance.

-------------------------------------- Lab Assignments ------------------------------------

Question 1: Write a program to store your friends' contact information by declaring and using a named structure called `person`. The struct `person` must include the following members:

- Name (string with at least 80 characters), e.g., "Alice", "Bob", "Mallory"
- Phone (string with at least 10 characters), e.g., "0817635249", "0868739293"

Create two variables from the struct `person` and populate them by reading the user input from the keyboard. After entering all the information, print out the values of the two struct variables sorted by their either their names from A → Z or their phone number from 0 → 9.

Expected Output:

```
Enter name#1: Alice
Enter phone#1: 0819998888
Enter name#2: Mallory
Enter phone#2: 0810001111
Sort by: name
Alice 0819998888
Mallory 0810001111
```

```
Enter name#1: Alice
Enter phone#1: 0819998888
Enter name#2: Mallory
Enter phone#2: 0810001111
Sort by: phone
Mallory 0810001111
Alice 0819998888
```

Bonus Question: You will upgrade the Monster Hunter mini game introduced in Week 12 to version 2.0 by following the requirements below:

- There are three monsters named `bone`, `snot` and `plague.`
- The player has an HP score of 100.
- Each monster has a name, an attack power, and an HP score.
  - Name: bone
    - Attack: 20
    - HP: 50
  - Name: snot
    - Attack: 30
    - HP: 100
  - Name: plague
    - Attack: 50
    - HP: 70
- Use the named structure called `monster` shown below for storing the monster data.

```
struct monster
{
    char name[80];
    int attack;
    int hp;
};
```

- Create an array called `monsters` for storing the three monsters and automatically populate the array with the given monsters' information.
- When the game starts, ask the player to input a **full name** of the monster and the player's **attack power (ATK).**
- Deduct the HP of the named monster according the given attack power.

- If the monsters are not killed, they also attack the player back based on their attack power.
- The game stops when all the monsters are killed, or the player is killed.
- **Note:** if the ATK > HP, the remaining HP will be 0.

Expected Output

```
Enter the monster's name: bone
Enter attack power from 1 to 100: 30
Monster: bone
HP: 20
Your HP: 80
Enter the monster's name: bone
Enter attack power from 1 to 100: 30
Monster: bone
HP: 0
Your HP: 80
Enter the monster's name: snot
Enter attack power from 1 to 100: 100
Monster: snot
HP: 0
Your HP: 80
Enter the monster's name: plague
Enter attack power from 1 to 100: 70
Monster: plague
HP: 0
Your HP: 80
You won
```

```
Enter the monster's name: plague
Enter attack power from 1 to 100: 10
Monster: plague
HP: 60
Your HP: 50
Enter the monster's name: snot
Enter attack power from 1 to 100: 10
Monster: snot
HP: 90
Your HP: 20
Enter the monster's name: bone
Enter attack power from 1 to 100: 10
Monster: bone
HP: 40
Your HP: 0
You lose
```