# Homework # 7

## 01286131 Object-Oriented Programming

## Software Engineering Program,

## Department of Computer Engineering,

## School of Engineering, KMITL

By

65011277 Chanasorn Howattanakulphong

## Object-Oriented Programming Homework #7 Mar 3rd, 2023 Review Exercises

**1.** Write functions and programs that manage `Person` records, verify correctness of all functions with test programs and ensure that all programs run as expected.

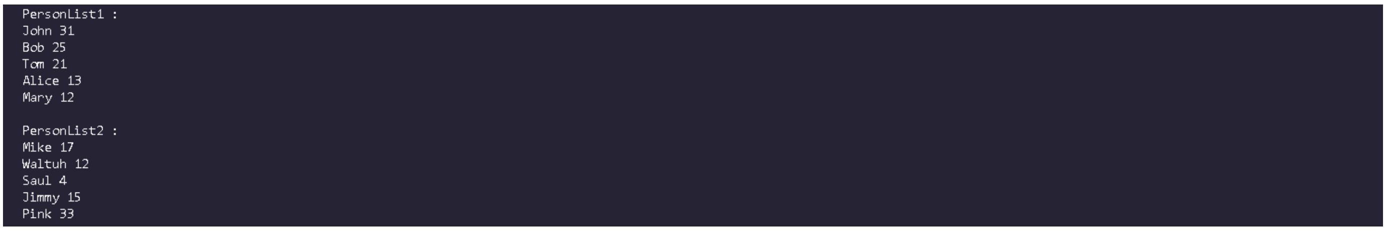
**1.1)** Define `Person` structure with **name** (string) and **age** (integer) as its data fields. Use the structure to do the following:

- Write a program to sort the list of persons by **name** in **ascending order**
- Write a program to sort the list of persons by **name** in **descending order**
- Write a program to sort the list of persons by **age** in **ascending order** Write a program
- to sort the list of persons by **age** in **descending order**

**1.2)** Use **Person** structure in **1.1)**, to do the following:

- Write a function `combine_fields(names, ages)`, that creates the list of persons out of the list `names` and the list `ages`
- Write a function `extract_names(person_list)`, that extracts the **name** list from the list of persons
- Write a function `extract_ages(person_list)`, that extracts the **age** list from the list of persons

**1.3)** Modify functions and programs from **1.2)** to use free store memory to store the list of person **without using the C++ standard library**. Define additional support types as necessary.



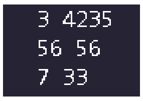
```
PersonList1 :
John 31
Bob 25
Tom 21
Alice 13
Mary 12

PersonList2 :
Mike 17
Waltuh 12
Saul 4
Jimmy 15
Pink 33
```

**2.** Write functions and programs that manage `Point` records, verify correctness of all functions with test programs and ensure that all programs run as expected.

**2.1)** Define `Point` structure with **x** (number) and **y** (number) as its data fields. Use the structure to do the following:

- Write a function `create_point_list(xlist, ylist)`, that creates the list of points out of the list `xlist` and the list `ylist`
- Write a function `split_point_fields(point_list)`, that extracts both the **x** list and the **y** list from the list of points

**2.2)** Modify functions and programs from **2.1)** to use free store memory to store the list of point **without using the C++ standard library**. Define additional support types as necessary.



```
3 4235
56 56
7 33
```