\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**CSC249 data structure and algorithms**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

CHAPTER 1B LAB **BASIC PYTHON PROGRAMMING (PART B)**

# Objectives

- Define functions to structure code

- Use built-in data structures such as strings, files, lists, tuples, and dictionaries

- Define classes to represent new types of objects

# Instruction and Problems

Write a Python program for each of the problems in this lab. Please use PyCharm to type and test your programs. Submit the Python files to Blackboard for credit.

## Program 1

Write a program for course registration. Students can use this program to add and drop courses. There should be a loop in the program that tells the user to enter A to add course, D to drop course or E to exit. If A is chosen, ask the user to enter the course to add. If the user is already registered in that course, display the message “You are already registered in that course”; otherwise, add the course to the user’s course list, display the message “Course added”, sort the list and display the list. If the user chooses D, ask the user to enter the course to drop. If the user is not registered in that course, display the message “You are not registered in that course”; otherwise, remove the course from the user’s course list, display the message “Course dropped” and display the list. The loop will stop when the user enters E.

The following is an example:

Enter A to add course, D to drop course, and E to exit: A

Enter course to add: CSC151

Course added

Courses registered: ['CSC151']

Enter A to add course, D to drop course, and E to exit: A

Enter course to add: CSC134

Course added

Courses registered: ['CSC134', 'CSC151']

Enter A to add course, D to drop course, and E to exit: A

Enter course to add: CSC139

Course added

Courses registered: ['CSC134', 'CSC139', 'CSC151']

Enter A to add course, D to drop course, and E to exit: D

Enter course to drop: CSC134

Course dropped

Courses registered: ['CSC139', 'CSC151']

Enter A to add course, D to drop course, and E to exit: A

Enter course to add: CSC151

You are already registered in that course

Enter A to add course, D to drop course, and E to exit: A

Enter course to add: CSC121

Course added

Courses registered: ['CSC121', 'CSC139', 'CSC151']

Enter A to add course, D to drop course, and E to exit: E

Save your Python program in a file named **Lab01BP1.py**. Submit the file to Blackboard for credit.

## Program 2

Energy consumption is measured in units of kilowatt hours (kWh). The more kWh a household use in a month, the higher the energy bill. Residential customers pay $0.12 per kWh. Business customers pay $0.16 per kWh. Write a program to calculate energy charge. You must write and use the following functions.

1. A main function: Call the value returning function get\_user\_input, which returns kWh used and customer type. Pass the return values to the value returning function bill\_calculator as two arguments. Display the return value of bill\_calculator.
2. A get\_user\_input function: This function has no parameter. It asks the user to enter number of kWh used. Use an input validation loop to ensure that kWh used is not negative. Also ask the user to enter customer type (enter R for residential or B for business). Convert lowercase letter to uppercase. Use an input validation loop to ensure that customer is either R or B. Return kWh used and customer type.
3. A bill\_calculator function: This function has two parameters to receive number of kWh used and customer type. Calculate and return the energy charge.

The following is an example.

Enter kilowatt hours used: -100

kWh cannot be negative.

Enter kilowatt hours used: 100

Enter R for residential customer, B for business customer: R

Please pay this amount: $12.00

Save your Python program in a file named **Lab01BP2.py**. Submit the file to Blackboard for credit.

## Program 3

The Payroll Department keeps a list of employee information for each pay period in a text file. The following is the content of the file for last week:

Davis 10.50 24.0

Bullock 12.75 32.0

Evans 13.00 27.5

Jones 11.25 10.50

The name of this file is payroll\_data.txt. Each line has three fields: last name, hourly pay rate and hours worked. Write a program that reads data from this file, calculates the total pay and displays name, hourly rate, hours worked and total pay for each employee.

Expected output:

Name Rate Hours Total Pay

Davis 10.50 24.00 252.00

Bullock 12.75 32.00 408.00

Evans 13.00 27.50 357.50

Jones 11.25 10.50 118.12

Save your Python program in a file named **Lab01BP3.py**. Submit the file to Blackboard for credit.

# Grading rubric

Program 1:

Add course [10 points]

Drop course [10 points]

Sort and display course list [10 points]

Other statements [10 points]

Program 2:

main function [10 points]

get\_user\_input function [10 points]

bill\_calculator function [10 points]

Program 3:

Reading data [15 points]

Calculating total pay [10 points]

Displaying output [5 points]