

CENG241 Labwork 5

1. Implement the following **Course** class:

```
Course
+ Course()
+ Course(double, double, double)
+ calculate(): void
- mtG, hwG, finG: int
- mtW, hwW, finW: double
- name: string
- letterGrade: string
```

In addition; add set/get functions for midterm, homework, final, name variables and only get function for letterGrade variable.

Default constructor should set midterm, homework and final weights to 0.30%, 0.20% and 0.50% respectively, while overloaded constructor should be able to obtain these values from parameters.

Create two objects from this Course class in your main function; one using default constructor and one using overloaded constructor. Read grades from the keyboard for both courses and calculate letterGrade by calling the calculate() function. Print both courses on screen using get functions in a table. You may refer to university regulations for letter grade ranges.

Enter weights for second course: 0.40 0.15 0.45

Enter first course name: OOP101

Enter first course grades: 67 85 78

Enter second course name: OOP201

Enter second course grades: 57 65 26

Course	Midterm	Homework	Final	Letter
OOP101	67	85	78	CB
OOP201	57	65	26	FD

2. Write a menu-drive C++ address book program. Your program should be able to display the address book and add new or remove existing contacts from/to it. In order to do so, develop the following classes first:

```
Contact
+ Contact()
+ Contact(string, string)
+ print(string): void
- name: string
- number: string
```

```
AddressBook
+ AddressBook()
+ add(Contact): bool
+ remove(string): bool
+ print(): void
- search(string): int
- contacts: Contact[]
- nrofcontacts: int
```

Remarks:

- (a) Implement **get/set** functions for private variables in Contact class.
- (b) Contact **default constructor** sets name and number to ("Unknown", "0").
- (c) Contact **print()** function takes a string parameter. Its values can be "light" or "headers". Light mode simply prints name and number next to each other on the same line, while headers mode prints in "Name: abc", "Number: 123" format on different lines.

- (d) AddressBook **add()** function should check if a contact with the same name already exists or not and returns true or false depending on the operation.
- (e) AddressBook **remove()** function should check if a contact with the same name already exists or not and returns true or false depending on the operation.
- (f) AddressBook **search()** function should check if a contact with the given name exists and return its position in the array (-1 otherwise). Keep the function private; use it for helping print, add, remove functions.
- (g) You don't need to implement dynamic arrays yet; so you may use a fixed size **Contact** array and use **nrofcontacts** variable to control its size for the moment.
- (h) You may use additional variables/functions in both classes.

```
1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit
Enter your choice: 1
No contacts available.
```

```
1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit
Enter your choice: 2
Enter name: Sid
Enter number: 0535
Contact added.
```

```
1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit
Enter your choice: 2
Enter name: Scrat
Enter number: 0557
Contact added.
```

```
1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit
Enter your choice: 2
Enter name: Diego
Enter number: 0506
Contact added.
```

```
1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit
Enter your choice: 1
Name           Number
Sid            0535
Scrat          0557
Diego          0506
```

1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit

Enter your choice: 3

Enter name: Scrat

Contact removed.

1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit

Enter your choice: 1

Name	Number
------	--------

Sid	0535
-----	------

Diego	0506
-------	------

1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit

Enter your choice: 3

Enter name: Scrat

Contact is not available.

1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit

Enter your choice: 2

Enter name: Sid

Enter number: 0535

Contact already exists.

1. Display Address Book
2. Add New Contact
3. Remove Contact
4. Exit

Enter your choice: 4

Goodbye.