ITI 1121. Introduction to Computer Science II

Laboratory 3

Winter 2015

Objectives

- Further understanding of object oriented programming

Declaring new types

Class declarations are one of the ways of defining new types in Java. For this laboratory, we will create two classes that both represent time values for a 24 hours period. The valid operations are as follows.

- **int getHours():** returns the number of hours
- **int getMinutes():** returns the number of minutes
- **int getSeconds():** returns the number of seconds
- **String toString():** returns a String representation
- **boolean equals(Time other):** returns true if and only if **other** designates an object that has the same content as this one
- **void increase():** increases this time by one second
- **Time plus(Time other):** returns a new time object who value represents the sum of this time and that of **other**
- **boolean before(Time other):** returns true if and only if this time object represents a time value that is before that of **other**

Note: at the end of this laboratory, you will submit the content of a directory called **l3_123456**, where 123456 has been substituted by your student id. You might as well create the directory now and save all the files that you will be creating into that directory.

1 Time1

Consult the declaration of the class **Time1**.

- Time1.java

Modify the declaration of the class Time1 and implement the following two methods.

- **Time1 plus(Time1 other):** returns a new time object who value represents the sum of this time and that of other
- **boolean before(Time1 other):** returns true if and only if this time object represents a time value that is before that of other

Create a new class called **Utils1** containing the following two class methods:

- **boolean isIncreasing(Time1[] ts)**: returns true if and only if the elements of the array designated by the formal parameter **ts** are in increasing order of time.
- **void main(String[] args)**:
    1. The method obtains 3 integer values from the command line (see Laboratory 1)
    2. It declares a reference variable **t** of type **Time1**
    3. It creates a **Time1** object using the integer values read from the command line and assigns the reference to the variable **t**
    4. The method declares a reference variable **times** to designate an array of time objects
    5. Create an array for 5 time objects, store the reference of that array into the variable **times**
    6. Assign **t** to **times[0]**
    7. For positions 1 to 4 of the array designated by **times**, **times[i]** is **times[i-1]** plus **t**.
    8. Finally, print the result of calling the method **isIncreasing** using **times**

2 Time2

Let's consider an alternative way of representing time values. Create a new class called **Time2**. Instead of having three instance variables (hours, minutes and seconds) like **Time1**, **Time2** has a single instance variable, **timeInSeconds**. For instance, the time value 2:5:10 will be represented as

```
timeInSeconds = 10 + 5*60 + 2*60*60 = 10 + 300 + 7200 = 7510
```

The maximum value for the variable **timeInSeconds** is the total number of seconds for a 24 hours period minus 1.

- Write a class called **Time2** that implements all the methods that are implemented by the class **Time1**.

Note: Just like **Time1**, the constructor of the class **Time2** has three parameters, **hours**, **minutes**, and **seconds**.

Copy the file **Utils1.java** to **Utils2.java** and make all the necessary changes so that **Utils2** uses **Time2** objects instead of **Time1** objects.

3 Quiz (1 mark)

Knowing that the instance variable **timeInSeconds** of the class **Time2** has been declared to be **private**. The following implementation of the method **equals**, within the class **Time2**, is valid. **True** or **False**.

```
public boolean equals(Time2 other) {
    return timeInSeconds == other.timeInSeconds;
}
```

Answer the question on Blackboard Learn:

- https://uottawa.blackboard.com/

Last Modified: January 29, 2015