

CST 183 Java Programming Programming Assignment 2 (30 pts)

Purpose To build a well-documented Java application focused on test-driven development, basic input/output, integer multiplication, division, and modulus operations, uses JGrasp, and converts military time to standard.

Requirements Write a Java program that can serve as an *ending time* calculator. The user enters the starting time in hours and minutes, a duration in total minutes, and your program will calculate and display the ending time (as hours:minutes).

For example, if an event starts at 2 30 and lasts 125 minutes, it will end at 4 35

Note: To simplify the problem, assume military time (0..23) rather than standard time, in which you would need to worry about a.m. and p.m.

Step 1: Begin by developing test cases for this program.

Ending time should be expressed in military time (0..23)

For example: 1:30 p.m. is expressed as 13:30 military time

Test Cases				
All times are in military time				
Start Hours	Start Minutes	Duration (in minutes)	Ending Hour (military time 0..23)	Ending Minute (0..59)
1	30	120		
11	45	60		
12	40	90		
10	0	240		

Step 2: Think about the math. Use integer multiplication, division, and modulus operators.

The **algorithm** to solve this problem on a computer is not straight-forward. Here is a solution:

1. **Convert** the starting time in hours and minutes to the equivalent **total minutes**
 - For example: 2 hours and 30 minutes would be 150 total minutes
 - For example: 5 hours and 15 minutes would be 315 total minutes

- Hint: use multiplication and addition

Write down the **math formula** needed to calculate the starting time (in total minutes) given the start hours and start minutes.

startingTime (in total minutes) = _____

2. **Add** the duration (in minutes) to this starting time (in total minutes), giving you the ending time in total minutes
 - For example: a start time of 150 (in total minutes) plus 125 minutes duration gives you an ending time of 275 total minutes
 - For example: a start time of 315 (in total minutes) plus 10 minutes duration gives you an ending time of 325 total minutes

Write down the **math formula** needed to calculate the ending time (in total minutes) given the starting time (in total minutes) and the duration (in minutes).

endingTime (in total minutes) = _____

3. Convert the ending time in total minutes back to hours and minutes with / and % integer operators and the fact that one hour is **60** minutes
 - For example, 63 minutes is 1 hour and 3 minutes
 - For example, 125 minutes is 2 hours and 5 minutes
 - For example, 325 minutes is 5 hours and 25 minutes

Write down the **math formula** needed to calculate the ending hour and minutes given the ending time (in total minutes).

Note: at this point, **assume simple military time** for the end hours. So 12:50 plus 30 minutes would be 13:20

endingHour = _____

endingMinutes = _____

Step 3: Develop Logic Plan

Draw a basic flowchart or write pseudocode that details your logic plan for solving this problem and helps you plan the program. (Hint: Look at the code below)

Step 4: Code It

Use jGRASP or your IDE to write the Java program `EndingTime.java` that calculates the ending time (in hours and minutes), given a starting time (in hours

and minutes) and a duration (in minutes).

Below is the skeleton class code for `EndingTime`:

1. **Copy and paste** the skeleton class code into `EndingTime.java` to get the ball rolling.
2. Type in your program code
3. **Compile** your program and correct all the syntax errors
4. **Run** your program and compare your results to the test case values you hand calculated (in military hours).

```
//Written by: add your name and date
//Program Description: add description
//
import java.util.Scanner;

public class EndingTime
{
    public static void main(String [] args)
    {
        // Variable declarations
        // Hint: All variables need to be declared as integers

        // Create a Scanner object to read from the keyboard
        Scanner keyboard = new Scanner(System.in);

        // Get the starting time in hours and minutes

        // Get the duration time in minutes

        // Calculate the ending time

        // Display the output

    }
}
```

Sample session when executed:

```
Enter the starting time (in hours and minutes): 2 30
Enter the duration (in minutes): 125
Ending hour is 4
Ending minute is 35
```

Step 5: Kick it up a notch.

Convert military time to standard

Modify your program so that it converts the hours in military time (0..23) to hours in standard time (1..12).

- For example: 13 30 would be converted to 1 30
- 15 10 would be converted to 3 10
- You do not need to worry about AM or PM

Doing the *modulus* 12 conversion is not as straightforward as it may at first seem. Dividing by 12 will always returns a remainder value between 0 .. 11. Standard time hours are expressed as 1 .. 12.

A formula to perform the conversion is

$$h = ((h + 11) \bmod 12) + 1$$
 where h is military time hours and \bmod is the modulus operator %

1. Modify your program using the formula given to produce ending times using **standard time**
2. **Compile** your program and find and correct all the compiler syntax errors identified by the compiler
3. **Run** your program and compare your results to the test case values you hand calculated (regular hours)

Deliverables Submit the Java source code file (the .java file you completed) to the **Program 2 Dropbox** within the Delta eLearning System.

Create a hardcopy for turn-in and grading containing the following:

- a) Title Page
- b) Flow chart or pseudocode for program
- c) Source Code (copy and paste from your IDE)
- d) Output (copy and paste from your IDE).