**课程名称： Java EE平台技术**

**项目名称： JAVA编程（基础练习）**

一、**实验目的**

学生通过使用JAVA语言进行基本程序的开发，掌握JAVA通用IDE，练习类的封装使用、JAVA基本类库的使用、利用UML进行简单建模。

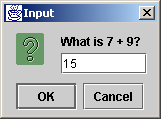
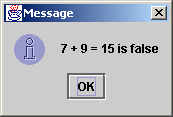
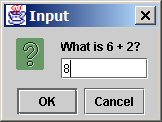
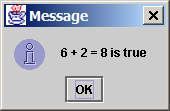
**二、实验内容**

Project #1 A Math Learning Tool (covers Chapter 2)

CSCI 1301 Introduction to Programming

Armstrong Atlantic State University

Develop a program that lets a first grader practice additions. The program randomly generates two single-digit integers and displays a question such as “What is 7 + 9?” to the student, as shown in Figure a. After the student types the answer in the input dialog box, the program displays a message dialog box to indicate whether the answer is true or false, as shown in Figure b.

(a) (b) (c) (d)

NOTE: Write the program without using conditional statements. Use only the materials you learned in Chapter 2.

Hint: Generate the random number using the System.currentTimeMillis() introduced in Chapter 2.

Submit the following items:

1. Analysis: Restate the problem in your own words (e.g., what is input, output if any, what needs to be computed, what data needs to be stored and their type).
2. Design: Clearly describe the steps to solve this problem using English or pseudo code.
3. Coding: Implement the solution in Java.
4. Testing: Make sure that the numbers appear random. Submit the screen shots of two sample runs.

Project #2 Tax Table (covers Chapter 3)

CSCI 1301 Introduction to Programming

Armstrong Atlantic State University

The United States federal personal income tax is calculated based on filing status and taxable income. There are four filing statuses: single filers, married filing jointly, married filing separately, and head of household. The tax rates for 2005 are shown in Table (a)-(d). If you are, say, single with a taxable income of $10,000, the first $7,300 is taxed at 10% and the other $2,700 is taxed at 15%. So your tax is 730 + 2700 \* 0.15.

**Schedule X — Single**

| **If taxable income is  over--** | **But not over--** | **The tax is:** |
| --- | --- | --- |
| $0 | $7,300 | 10% of the amount over $0 |
| $7,300 | $29,700 | $730 plus 15% of the amount over 7,300 |
| $29,700 | $71,950 | $4,090.00 plus 25% of the amount over 29,700 |
| $71,950 | $150,150 | $14,652.50 plus 28% of the amount over 71,950 |
| $150,150 | $326,450 | $36,548.50 plus 33% of the amount over 150,150 |
| $326,450 | no limit | $94,727.50 plus 35% of the amount over 326,450 |

**Schedule Y-1 — Married Filing Jointly or Qualifying Widow(er)**

| **If taxable income is over--** | **But not over--** | **The tax is:** |
| --- | --- | --- |
| $0 | $14,600 | 10% of the amount over $0 |
| $14,600 | $59,400 | $1,460.00 plus 15% of the amount over 14,600 |
| $59,400 | $119,950 | $8,180 plus 25% of the amount over 59,400 |
| $119,950 | $182,800 | $23,317.50 plus 28% of the amount over 119,950 |
| $182,800 | $326,450 | $40,915.50 plus 33% of the amount over 182,800 |
| $326,450 | no limit | $88,320.00 plus 35% of the amount over 326,450 |

**Schedule Y-2 — Married Filing Separately**

| **If taxable income is over--** | **But not over--** | **The tax is:** |
| --- | --- | --- |
| $0 | $7,300 | 10% of the amount over $0 |
| $7,300 | $29,700 | $730 plus 15% of the amount over 7,300 |
| $29,700 | $59,975 | $4,090 plus 25% of the amount over 29,700 |
| $59,975 | $91,400 | $11,658.75 plus 28% of the amount over 59,975 |
| $91,400 | $163,225 | $20,457.75 plus 33% of the amount over 91,400 |
| $163,225 | no limit | $44,160.00 plus 35% of the amount over 163,225 |

**Schedule Z — Head of Household**

| **If taxable income is over--** | **But not over--** | **The tax is:** |
| --- | --- | --- |
| $0 | $10,450 | 10% of the amount over $0 |
| $10,450 | $39,800 | $1,045 plus 15% of the amount over 10,450 |
| $39,800 | $102,800 | $5,447.50 plus 25% of the amount over 39,800 |
| $102,800 | $166,450 | $21,197.50 plus 28% of the amount over 102,800 |
| $166,450 | $326,450 | $39,019.50 plus 33% of the amount over 166,450 |
| $326,450 | no limit | $91,819.50 plus 35% of the amount over 326,450 |

Develop a program that prints a tax table for taxable income from $50,000 to $60,000 with intervals of $50 for all four statuses, as follows:

**taxable Single Married Married Head of**

**Income Joint Separate a House**

50000 9846 7296 10398 8506

50050 9859 7309 10411 8519...

59950 12532 9982 13190 1119260000 12546 9996 13205 11206

NOTE: the numbers in the preceding tax table are incorrect intentionally. You should print the correct values in the table.

Submit the following items:

1. Analysis: Restate the problem in your own words (e.g., what is input, output if any, what needs to be computed, what data needs to be stored and their type).
2. Design: Clearly describe the steps to solve this problem using English or pseudo code.
3. Coding: Implement the solution in Java.
4. Testing: Submit the screen shots for first 20 lines in the tax table.

Project #3 Least Common Multiple (covers Chapter 5)

CSCI 1301 Introduction to Programming

Armstrong Atlantic State University

Definition: The least common multiple (LCM) of two numbers is the smallest number that is a multiple of both. For example, the LCM for 8 and 12 is 24, for 15 and 25 is 75, and for 120 and 150 is 600.

Develop a program that prompts the user to enter two integers and finds their least common multiple.

To find the LCM of two numbers, first create a prime factor table for each number. The first column of the table consists of all the prime factors and the second column tracks the occurrence of the corresponding prime factor in the number. For example, the prime factors for 120 are 2, 2, 2, 3, 5, so the prime factor table for number 120 is shown as follows:



The prime factors for 150 are 2, 3, 5, 5, so the prime factor table for number 150 is shown as follows:



The LCM of the two numbers consists of the factors with the largest occurrence in the two numbers. So the LCM for 120 and 150 is , where 2 appears three times in 120, 3 one time in 120, and 5 two times in 150.

Submit the following items:

1. Analysis: Restate the problem in your own words (e.g., what is input, output if any, what needs to be computed, what data needs to be stored and their type).
2. Design: Clearly describe the steps to solve this problem using English or pseudo code. (Hint: The prime factor table can be represented using a two-dimensional array. Write a method named getPrimeFactors(int number) that returns a two-dimensional array for the prime factor table.)
3. Coding: Implement the solution in Java.
4. Testing: Test your program to find the LCM for (120, 150), (7, 14), (7, 8), (1, 2), and (345, 455)

Project #4 Design Classes (covers Chapter 6)

CSCI 1302 Introduction to Programming

Armstrong Atlantic State University

This project consists of two separate problems.

1. Design a class named Rectangle to represent a rectangle. The class contains:

* Two double data fields named width and height that specify the width and height of the rectangle. The default values are 1 for both width and height.
* A string data field named color that specifies the color of a rectangle. Hypothetically, assume that all rectangles have the same color. The default color is white.
* A no-arg constructor that creates a default rectangle.
* A constructor that creates a rectangle with the specified width and height.
* The accessor and mutator methods for all three data fields.
* A method named getArea() that returns the area of this rectangle.
* A method named getPerimeter() that returns the perimeter.

Draw the UML diagram for the class. Implement the class.

Write a test program that creates two Rectangle objects. Assign width 4 and height 40 to the first object and width 3.5 and height 35.9 to the second object. Assign color red to all Rectangle objects. Display the properties of both objects and find their areas and perimeters.

2. Design a class named Time. The class contains:

* Data fields hour, minute, and second that represents a time.
* A no-arg constructor that creates a Time object for the current time. (The data fields value will represent the current time)
* A constructor that constructs a Time object with a specified elapse time since the middle night, Jan 1, 1970 in milliseconds. (The data fields value will represent this time)
* Three get methods for the data fields hour, minute, and second, respectively.

Draw the UML diagram for the class. Implement the class. Write a test program that creates two Time objects (using new Time() and new Time(555550000)) and display their hour, minute, and second.

Hint: The current time can be obtained using System.currentTime(), as shown in Listing 2.8, ShowCurrentTime.java. The other constructor sets the hour, minute, and second for the specified elapse time. For example, if the elapse time is 555550000 milliseconds, the hour is 10, the minute is 19, and the second is 10.

**三、实验环境**

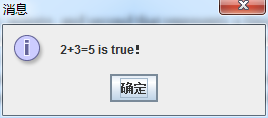
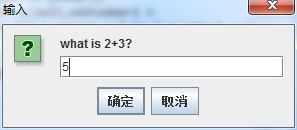
* 1. Windows XP，Windows 7
  2. Eclipse
  3. NetBean

**四、实验结果**

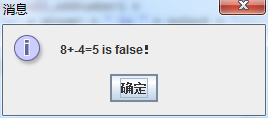
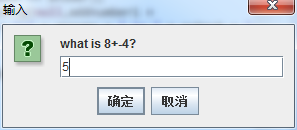
Project 1:

随机生成两个数，看用户输入的和与真实和是否相等：

1、正确：

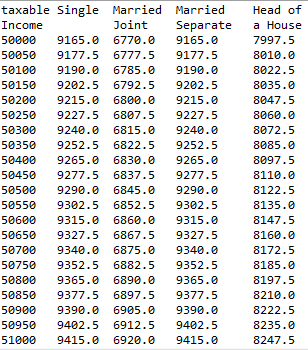


2、错误：



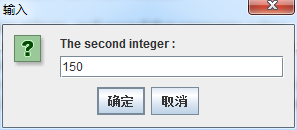
Project 2:

以下为工资50000~60000（每次增长50）前20名对应的四种类型的税收：

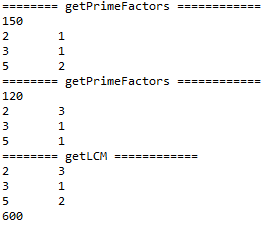


Project 3:

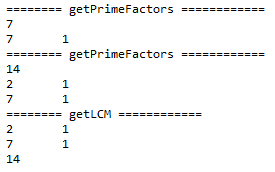
用户输入两个整型数，程序得出两个整型数的最小公倍数：



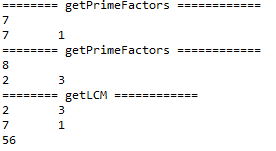
a、(120,150)：



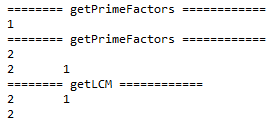
b、(7,14)：



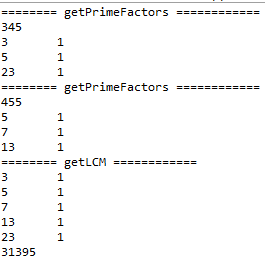
c、(7,8)：



d、(1,2)：



e、(345,455):



Project 4:

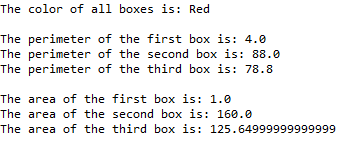
Rectangle:

第一个矩形为无参构造函数形成的矩形（width:1,height:1,color:white）；

第二个矩形为有参构造函数形成的矩形（width:4,height:40,color:white）；

第三个矩形为有参构造函数形成的矩形（width:3.5,height:35.9,color:white）；

设置所有填充颜色皆为：red。



Time:

第一个时间为无参构造函数形成的时间，它显示了从1970-01-01 00:00:00.000到现在所经过的时间（只显示小时、分钟、秒）；

第二个时间为有参构造函数形成的时间（new Time(555550000)）。



**五、附录**

调试心得：

1、Java有很强大的库，许多内容可引用它本身的库去做，例如一些特定的弹窗、对System.currentTimeMills()时区造成的时差的处理等；

2、在求最小公倍数时注意质数的判断和质数数组的创建，在引入二维数组时位置要处理好，且当把一个数拆成若干个质数后，记得跳出判断是否整除的循环，否则当商为“1”且不退出循环后，可能会除至数组中未存数的部分，这时除以0会有错误；

3、构建矩形、时间类的时候，要注意有参与无参函数的创建，且最好加入对类中变量的操作的函数（set/get）。