CSCE 156 – Computer Science II
Lab 02 - Conditionals & Loops

Dr. Chris Bourke

Prior to Lab

Review this laboratory handout prior to lab.

For Java:

1. Read if-then-else tutorial:
   http://download.oracle.com/javase/tutorial/java/nutsandbolts/if.html

2. Read switch/case tutorial:
   http://download.oracle.com/javase/tutorial/java/nutsandbolts/switch.html

3. Read for loop tutorial:
   http://download.oracle.com/javase/tutorial/java/nutsandbolts/for.html

4. Read while/do while loop tutorial:
   http://download.oracle.com/javase/tutorial/java/nutsandbolts/while.html

For PHP:

1. Refer to the Control Structures section in the PHP Manual:
   http://www.php.net/manual/en/language.control-structures.php

Lab Objectives & Topics

Following the lab, you should be able to:

- Use if-then-else statements to control the logical flow of the program.

- Use switch-case statement to control the logical flow of the program.

- Use for/while loops to implement repetition statements in your program.

- Write complex programs that require conditional logical statements and or loops.

Peer Programming Pair-Up

To encourage collaboration and a team environment, labs will be structured in a *pair programming* setup. At the start of each lab, you will be randomly paired up with another student (conflicts such as absences will be dealt with by the lab instructor). One of you will be designated the *driver* and the other the *navigator*.

The navigator will be responsible for reading the instructions and telling the driver what to do next. The driver will be in charge of the keyboard and workstation. Both driver and navigator are responsible for suggesting fixes and solutions together. Neither the navigator nor the driver is "in charge." Beyond your immediate pairing, you are encouraged to help and interact and with other pairs in the lab.

Each week you should alternate: if you were a driver last week, be a navigator next, etc. Resolve any issues (you were both drivers last week) within your pair. Ask the lab instructor to resolve issues only when you cannot come to a consensus.

Because of the peer programming setup of labs, it is absolutely essential that you complete any pre-lab activities and familiarize yourself with the handouts prior to coming to lab. Failure to do so will negatively impact your ability to collaborate and work with others which may mean that you will not be able to complete the lab.

Getting Started

Clone the project code for this lab from GitHub in Eclipse using the URL, https://github.com/cbourke/CSCE156-Lab02-ConditionalsLoops. Refer to Lab 01 for instructions on how to clone a project from GitHub.

For those with prior Java experience, do the PHP section. For those without prior Java experience, do the Java section.

PHP

Conditionals & Loops

PHP provides standard control structures for conditionals and repetition. Specifically, PHP provides the usual if-then-else statements and while, for, and do-while loops. The syntax for these control structures should look familiar; some examples:

```php
1   if(condition1) {
2      //DO SOMETHING
3   } else if(condition2) {
4      //DO SOMETHING ELSE
5   } else {
6      //OTHERWISE
7   }
8
9   for($i=0; $i<$n; $i++) {
10     //DO SOMETHING
11  }
12
13  $i=0;
14  while($i<$n) {
15     //DO SOMETHING
16     $i++;
17  }
18
19  $i=0;
20  do{
21     //DO SOMETHING
22     $i++;
23  } while($i<$n);
```

In addition, PHP provides a foreach-loop for iterating over elements in an array. This is not just for convenience: in PHP arrays are associative so they are not necessarily indexed 0 thru n - 1; arrays may not even be indexed with integers! Instead, array should be considered to be a collection of key-value pairs. The following examples illustrate the foreach loop's usage.

```php
1   foreach($array as $value) {
2     print "$value \n";
3   }
4
5   foreach($array as $key => $value) {
6     print "The key $key maps to the value $value\n";
7   }
```

Activities

Sum of Natural Numbers

Natural numbers are the usual counting numbers; 1, 2, 3, ... In this exercise you will write several loops to compute the sum of natural numbers 1 thru n where n is read from the command line. You will also write a foreach loop to iterate over an array and process data.

1. Open the `natural.php` source file. The code to read in n has already been provided for you. An array mapping integer values 1 thru 10 to text values has also been created for you.

2. Write a for-loop and a while-loop to compute the sum of natural numbers 1 thru n and output the answer.

3. Write a foreach loop to iterate over the elements (key/value pairs) of the `$oneToTen` array. As you iterate over the elements you should sum the keys and concatenate the values to formulate the following string (which you should output at the end of the for-loop):

```
one + two + three + four + five + six + seven + eight + nine + ten = 55
```

4. Hand in your program using the webhandin and use the webgrader to verify your program works correctly.

Child Tax Credit

When filing for federal taxes, a credit is given to tax payers with dependent children according to the following rules. The first dependent child younger than 18 is worth a $1000.00 credit. Each dependent child younger than 18 after the first child is worth a $500 tax credit each. You will complete a PHP script to output a table of dependent children, how much each contributes to a tax credit, and a total child tax credit. Your table should look something like the following.

```
Child           Amount
Tommy (14)      $1000.00
Richard (12)    $500.00
Harold (21)     $0.00
Total Credit:   $1500.00
```

1. Open the `Child.php` and `ChildCredit.php` script files

2. The `Child` class has already been defined and included in the `ChildCredit.php` script. Note how the `Child` class is used; several instances of children have been created and placed into an array.

3. Write code to iterate over the array, compute the child tax credits and output a table similar to the one above. Note: to call a method on an instance of the `Child` class, use the following syntax: `$kid->getAge()`

4. Answer the questions in your worksheet and demonstrate your working code to a lab instructor.

Advanced Activity (Optional)

Modify the Child Tax Credit program to output the data in a well-formatted HTML table. Demonstrate your dynamic webpage to a lab instructor.

Java

Conditionals & Loops

Java provides standard control structures for conditionals and repetition. Specifically, Java provides the usual if-then-else statements and while, for, and do-while loops. The syntax for these control structures should look familiar; some examples:

```java
if(condition1) {
  //DO SOMETHING
} else if(condition2) {
  //DO SOMETHING ELSE
} else {
  //OTHERWISE
}

for(int i=0; i<n; i++) {
  //DO SOMETHING
}

int i=0;
while(i<n) {
  //DO SOMETHING
  i++;
}

int i=0;
do{
  //DO SOMETHING
  i++;
} while(i<n);
```

In addition, Java provides a foreach-loop, also referred to as an *enhanced for-loop*, for iterating over collections (classes that implement the `Iterable` interface) or elements in an array. This feature is mostly for convenience. The following examples illustrate this loop's usage.

```java
String arr[] = new String[10];
...
for(String s : arr) {
  System.out.println(s);
}
```

Activities

Sum of Natural Numbers

Natural numbers are the usual counting numbers; 1, 2, 3, ... In this exercise you will write several loops to compute the sum of natural numbers 1 thru n where n is read from the command line. You will also write an enhanced for-loop to iterate over an array and process data.

1. Open the `Natural.java` source file. The code to read in n has already been provided for you. An array mapping integer values 1 thru 10 to text values has also been created for you.

2. Write a for-loop and a while-loop to compute the sum of natural numbers 1 thru n and output the answer.

3. Write a foreach loop to iterate over the elements (key/value pairs) of the `zeroToTen` array. As you iterate over the elements, concatenate each string, delimited by a single space to a result string and print the result at the end of the loop. Your result should look something like the following:

   ```
   zero + one + two + three + four + five + six + seven + eight + nine + ten = 55
   ```

4. Hand in your program using the webhandin and use the webgrader to verify your program works correctly.

Child Tax Credit

When filing for federal taxes, a credit is given to tax payers with dependent children according to the following rules. The first dependent child younger than 18 is worth a $1000.00 credit. Each dependent child younger than 18 after the first child is worth a $500 tax credit each. You will complete a Java program to output a table of dependent children, how much each contributes to a tax credit, and a total child tax credit. Your table should look something like the following.

```
Child           Amount
Tommy (14)      $1000.00
Richard (12)    $500.00
Harold (21)     $0.00
Total Credit:   $1500.00
```

1. Open the `Child.java` and `ChildCredit.java` source files

2. The `Child` class has already been implemented for you. Note how the `Child` class is used; several instances of children have been created and placed into an array.

3. Write code to iterate over the array, compute the child tax credits and output a table similar to the one above. Note: to call a method on an instance of the `Child` class, use the following syntax: `kid.getAge()`

4. Answer the questions in your worksheet and demonstrate your working code to a lab instructor.

Advanced Activity (Optional)

Use the `String.format()` method to reformat the output of the Child Tax Credit program to print every piece of data in its own column.