Lab10
Java Branching and Looping Statements

Submission Details

In this lab, you are required to write **ONE** Java program to solve the given problem shown in the section "Lab Exercise". To make the program implementation easier, you are suggested to write your program using **eclipse** instead of doing it directly using paper & pencil. After you have completed the implementation using eclipse, submit your program file (i.e. CrossTheRiver.java) to the assignment page of Canvas. The deadline would be **ONE** week after your lab section is conducted. You are reminded to double-check your solution to verify everything in correct before submission.

Note: You are required to put your name, student ID and lab section at the beginning of your source file. Also, please DON'T ZIP your files for submission. Instead, you should upload .java file to the Canvas.

Objective

The objective of this lab is to reinforce your programming concepts, especially Java branching and iterative (looping) statements taught in the last few lectures.

Software Required

The following software is required for this lab.

1. Java Development Kit Version 1.8
2. An Integrated Development Environment, e.g. eclipse

Introduction

In this lab, you will practice how to solve problems using branching and looping / iterative statements. For instances, using if, if-else and while loop. As usual, you will be given a brief review on these topics before going through the details of the lab exercise.
(Note: You are suggested to spend more time on the lab exercise. Since that should give you a thorough revision on what you should have learnt in class.)

Background

1. Do checking using branching statements:

- **Syntax:**

```
// If <boolean expression> is evaluated to true, <statement> is executed.
// Otherwise, just skip it.
if(<boolean expression>)
   <statement>;

// If <boolean expression> is evaluated to true, <statement 1>, <statement 2>,
// …,    are executed. Otherwise, just skip them.
if(<boolean expression>)
{
   <statement 1>;
   <statement 2>;
   // …
}

// If <boolean expression> is evaluated to true, <statement A> is executed.
// Otherwise, <statement B> is executed.
if(<boolean expression>)
   <statement A>;
else
   <statement B>;

// If <boolean expression> is evaluated to true, <statement A1>, <statement A2>,
//    …,    are executed. Otherwise, <statement B1>, <statement B2>, …, are
// executed.
if(<boolean expression>)
{
   <statement A1>;
   <statement A2>;
   // …
}
else
{
   <statement B1>;
   <statement B2>;
   // …
}
```

- **Syntax:**

```
// switch <expression> matches a case, all the statement(s) of that case is/are executed
// until a break statement is found. If none of the case matches, the statement(s) at
// the default case is/are executed.
switch(<expression>)
{
   case <constant a>:
        <statement a1>;
        // …
        break;
   // …
   default:
        <statement c1>;
        // …
}
```

2. **Executing statements repeatedly using looping statements:**

Syntax:

```
// If <boolean expression> is evaluated to true, <statement> is executed and this
// process is repeated until the <boolean expression> is evaluated to false.
while(<boolean expression>)
   <statement>;

// If <boolean expression> is evaluated to true, <statement 1>, <statement 2>,
// …,    are executed and this process is repeated until the <boolean expression> is
// evaluated to false.
while(<boolean expression>)
{
   <statement 1>;
   <statement 2>;
   // …
}
```

3. **Display information on screen:**

Syntax:

```
// Print data on screen and the cursor will stay at the end of the string printed
System.out.print(<data> [+ <data>]);
// Print the data on screen and move the cursor to the beginning of the next line
System.out.println(<data> [+ <data>]);
// Print nothing, but move the cursor to the beginning of the next line
System.out.println(“”);
```

Lab Exercise

In this lab, you will be asked to solve a problem using Java. Please refer to the problem specification below for the requirements.

Question:

Write a Java program "CrossTheRiver.java" to implement the game "**Crossing the River**".

1. At the start of the game, there are **THREE** cannibals and **THREE** missionaries on one side of a river.
2. Your task is to move all cannibals and missionaries to the other side of the river using boat.
3. The boat can only take **ONE** or **TWO** people each time.
4. When there are more cannibals than missionaries on one side of the river, they eat them and you lose the game.

The expected output of your program should be as follows:

Sample output 1:

```
Crossing the River
---------------------------------
Round 0:
Left           Right
MMMCCC
-------
\-----/
First passenger (c for cannibal, m for missionary): h
Illegal input!
First passenger (c for cannibal, m for missionary): m
Second passenger (c for cannibal, m for missionary, n for none): c
---------------------------------
Round 1:
Left           Right
MMCC           MC
               -------
               \-----/
First passenger (c for cannibal, m for missionary): c
Second passenger (c for cannibal, m for missionary, n for none): c
Illegal input!
Second passenger (c for cannibal, m for missionary, n for none): n
---------------------------------
Round 2:
Left           Right
MMCCC          M
-------
\-----/
Missionaries eaten by cannibals! You lose!
```

Sample output 2:

```
Crossing the River
--------------------------------
Round 0:
Left           Right
MMMCCC
-------
\-----/
First passenger (c for cannibal, m for missionary): m
Second passenger (c for cannibal, m for missionary, n for none): c
--------------------------------
Round 1:
Left           Right
MMCC           MC
               -------
               \-----/
First passenger (c for cannibal, m for missionary): m
Second passenger (c for cannibal, m for missionary, n for none): n
--------------------------------
Round 2:
Left           Right
MMMCC          C
-------
\-----/
First passenger (c for cannibal, m for missionary): c
Second passenger (c for cannibal, m for missionary, n for none): c
--------------------------------
.........
.........
.........
--------------------------------
Round 16:
Left           Right
CC             MMMC
-------
\-----/
First passenger (c for cannibal, m for missionary): c
Second passenger (c for cannibal, m for missionary, n for none): c
--------------------------------
Round 17:
Left           Right
               MMMCCC
               -------
               \-----/
--------------------------------
Congratulations! You win the game in 17 rounds!
```

Marking Scheme

The marking of this exercise will be based on the following criteria.

Graded items	Weighting
1. Correctness of program (i.e. whether your code is implemented in a way according to the requirements as specified.)	50%
2. Presentation of the Java codes (i.e. whether the program is properly indented, how close you follow the common conventions as mentioned in class, etc.)	30%
3. Documentation (with reasonable amount of comments embedded in the code to enhance the readability)	20%
	100%

Be aware of plagiarism! DON'T copy the program file from your friends or classmates. If any identical copy is found, 5% of the coursework marks will be deducted for each of the file owner.

Program Submission Checklist

Before submitting your work, please check the following items to see you have done a decent job.

Items to be checked	☑ / ☒
1. Did I put my name, student ID and lab section at the beginning of all the source files as comment?	☐
2. Did I put reasonable amount of comments to describe my program?	☐
3. Are they all in .java extension and named according to the specification?	☐
4. Have I checked that all the submitted code are compliable and run without any errors?	☐
5. Did I zip my source files using Winzip / zip provided by Microsoft Windows? Also, did I check the zip file and see if it could be opened? ***(Only applicable if the work has to be submitted in zip format.)***	☐
6. Did I submit my lab assignment to Canvas before the deadline?	☐