## **Description**

For this project, you will be writing a program called HockeyScores that is able to use information about the results of a seven game hockey series to gather additional data about the series. There will be a string of seven hockey game scores (Ex: 04-01) with each score separated by commas. There will also be another string indicating the number of power play goals scored by each team in each game. This string will be formatted in the same way as the game scores.

This project is worth 7% of your final grade. We recommend that you take it, along with the other projects in the class, very seriously.

Note: 5 points of your grade is based on Coding Style. You will need to update the Starter Code to follow the standards. Use the "Run" button to check your Coding Style without using one of your 4 submissions.

## **Hockey Overview**

This assignment does not require previous knowledge of hockey. We provide a short overview of the sport as it is relevant to the assignment below.

Hockey is a sport with two teams and is typically played on ice. There is a goal on each end of the rink similar to other sports such as soccer. Each team typically has six players on the ice at a time with one player being the goaltender. The game is played with a puck and each player has a stick. A team is awarded a goal if a player is able to hit the puck into the opposing team's net using the stick. Only the goaltender is allowed to touch the puck with their hands, the other players must use their stick at all times. A player may be called for a penalty for several reasons such as tripping a player on the opposing team. If a penalty is called, the team that committed the penalty must play with one less player for the next few minutes (the amount of time typically ranges from 2-5 minutes). This is known as a "power play". A "power play goal" is a goal that is scored during a power play by the team with more players on the ice. For this assignment, we are focusing on a series of seven games. The team that wins the majority of the seven games is the winner. In other words, if a team wins four or more out of the seven games, they win.

To win a game, a team must score more goals than the other team after 60 minutes of play.

## **Instructions**

Download the Starter Code. The class already contains the prompts to accept the input and processing to identify team scores by game. There are also prompts to get the names of the two teams, and these names are stored in String variables.

* Do not modify the **starter code** sections. Scores are converted to integers using parseInt().
* The input strings will have exactly 7 games. In each game, the first score listed is the score for team 1 and the second score listed is the score for team 2. Similarly, for each game, in the power play string, the first number listed is the number of power play goals scored by team 1 and the second number is the number of power play goals scored by team 2.
* The range of valid possible scores is 00 to 99.
* You may assume that all game scores and power play scores are valid. In other words, valid numerical values will be entered for the game scores and power play goals. There will be no game where both teams score the same number of overall goals (though they may score the same number of power play goals). There are no ties in hockey. If the game is tied after 60 minutes, it continues until one of the teams scores.

Your solution must do the following:

* Identify the winner of the series and the number of games won by each team. When printing the number of games won by each team, the wins by the team that won the series should always be printed first even if the team that won is team 2. For example, print "The predators won the series by a score of 4-3" instead of "The predators won the series by a score of 3-4". The first number printed will always be 4.
* Print the total number of goals each team scored.
* Print the total number of power play goals each team scored.
* Print the total number of standard goals each team scored. A "standard goal" is any goal that is not scored on a power play.
* Print the maximum number of goals scored by either team in any of the games along with which team scored that number of goals and which game they scored them in. The first game is labeled as game 1, the second game is game 2, and the last game is game 7. If the maximum number of goals has been scored multiple times, print the team and game number for the first occurrence. In other words, if the maximum number of goals scored is 9 by team 1 in game 2 and team 2 also scored 9 goals in game 4, print team 1 and game 2.
* Print the number of shutouts each team recorded. A shutout is a game in which the opposing team does not score any goals.

Output examples are in the next section. Follow the same formats and be sure to include the prompts!

Note: If any of the printed values for goals or shutouts are 1, the word should not be plural. For example, a statement that indicates 1 shutout should read "The penguins recorded 1 shutout" instead of "The penguins recorded 1 shutouts". The only exception is that the wording for maximum number of goals will be the same regardless of whether the value is 1 or not.

## **Testing**

Sample Test 1

Welcome!

Enter the name of team 1.

[penguins]

Enter the name of team 2.

[red wings]

Enter hockey scores for seven games.

[04-02,03-09,06-05,01-00,02-00,04-01,03-05]

Enter the number of power play goals for both teams in each game.

[02-00,01-04,02-02,00-00,00-00,02-00,01-02]

The penguins won the series by a score of 5-2

The penguins scored 23 total goals

The red wings scored 22 total goals

The penguins scored 8 power play goals

The red wings scored 8 power play goals

The penguins scored 15 standard goals

The red wings scored 14 standard goals

The penguins recorded 2 shutouts

The red wings recorded 0 shutouts

The maximum number of goals scored was 9 by the red wings in game 2

Sample Test 2

Welcome!

Enter the name of team 1.

[blackhawks]

Enter the name of team 2.

[predators]

Enter hockey scores for seven games.

[02-03,01-07,08-02,01-03,02-00,01-02,05-01]

Enter the number of power play goals for both teams in each game.

[00-01,00-03,03-00,01-00,00-00,00-01,02-01]

Blackhawks

Predators

02-03,10-07,08-02,01-03,02-00,01-10,05-01

00-01,00-03,03-00,01-00,00-00,00-01,02-01

The predators won the series by a score of 4-3

The blackhawks scored 20 total goals

The predators scored 18 total goals

The blackhawks scored 6 power play goals

The predators scored 6 power play goals

The blackhawks scored 14 standard goals

The predators scored 12 standard goals

The blackhawks recorded 1 shutout

The predators recorded 0 shutouts

The maximum number of goals scored was 8 by the blackhawks in game 3

Note: Brackets [] indicate input.

Note: Match this output exactly. Spelling mistakes will result in lost points.

We have also included a program that will allow you to create and run output tests automatically in the Starter Code. This will make it easier for you to verify that each possible progression through your solution is correct. Take a look at RunLocalTest.java. There are many utility features and tools that you do not need to worry about at the moment, instead, focus on the test case (included below):

@Test(timeout = 1000)

public void test\_one() {

// Set the input

String input = "penguins" + System.lineSeparator() +

"red wings" + System.lineSeparator() +

"04-02,03-09,06-05,01-00,02-00,04-01,03-05" + System.lineSeparator() +

"02-00,01-04,02-02,00-00,00-00,02-00,01-02" + System.lineSeparator();

// Pair the input with the expected result

String expected = "Welcome!\nEnter the name of team 1.\nEnter the name of team 2.\n" +

"Enter hockey scores for seven games.\nEnter the number of power play goals for both teams in each game.\n" +

"The penguins won the series by a score of 5-2\nThe penguins scored 23 total goals\n" +

"The red wings scored 22 total goals\nThe penguins scored 8 power play goals\n" +

"The red wings scored 8 power play goals\nThe penguins scored 15 standard goals\n" +

"The red wings scored 14 standard goals\nThe penguins recorded 2 shutouts\n" +

"The red wings recorded 0 shutouts\nThe maximum number of goals scored was 9 by the red wings in game 2\n";

// Runs the program with the input values

receiveInput(input);

HockeyScores.main(new String[0]);

// Retrieves the output from the program

String output = getOutput();

// Trims the output and verifies it is correct.

output = output.replace("\r\n", "\n");

assertEquals("Ensure that your results match the format of the ones given in the handout!",

expected.trim(), output.trim());

}

Determine the team names, input scores, and number of power play goals you would like to use and add them to the input String, with each of the three input values separated by a /n (simulates pressing enter). Then, calculate the correct results to output and add them for expected. You can either download the program and run the main method or use the "Run" button on Vocareum to run the test. You can repeat this process for as many possible combinations as you would like to test.

### **Public Test Cases Note**

For many homeworks and projects, we will give you test cases that correspond to several of the ways we will be testing your program. But, we will not give you test cases for ALL of the ways we will be testing your program. You should think of other test cases to use that will fully test every aspect of every feature of your program. Just because your program passes all the test cases we give you does not mean that it is fully correct and will receive a score of 100.

public class HockeyScores {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Welcome!");

System.out.println("Enter the name of team 1.");

String teamOne = scanner.nextLine();

System.out.println("Enter the name of team 2.");

String teamTwo = scanner.nextLine();

System.out.println("Enter hockey scores for seven games.");

String scores = scanner.nextLine();

System.out.println("Enter the number of power play goals for both teams in each game.");

String powerPlay = scanner.nextLine();

scanner.close();

// The values of each of the scores are defined below, you should use these int variables

/\* The string has 7 scores so the format of the string is

\* scoreOneTeamOne-scoreOneTeamTwo,scoreTwoTeamOne-scoreTwoTeamTwo,scoreThreeTeamOne-scoreThreeTeamTwo,

\* ...

\*/

int currentScoreIndex = 0;

int scoreOneTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreOneTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreTwoTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreTwoTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreThreeTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreThreeTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreFourTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreFourTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreFiveTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreFiveTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreSixTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreSixTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreSevenTeamOne = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

currentScoreIndex += 3;

int scoreSevenTeamTwo = Integer.parseInt(scores.substring(currentScoreIndex, currentScoreIndex + 2));

// The values of each of the power play goals are defined below, you should use these int variables

/\* The string has the number of power play goals listed for both teams in each of the seven games so

\* the format of the string is

\* powerPlayOneTeamOne-powerPlayOneTeamTwo,powerPlayTwoTeamOne-powerPlayTwoTeamTwo,powerPlayThreeTeamOne-powerPlayThreeTeamTwo,

\* ...

\*/

int currentPowerPlayIndex = 0;

int powerPlayOneTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayOneTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayTwoTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayTwoTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayThreeTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayThreeTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayFourTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayFourTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayFiveTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlayFiveTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlaySixTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlaySixTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlaySevenTeamOne = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

currentPowerPlayIndex += 3;

int powerPlaySevenTeamTwo = Integer.parseInt(powerPlay.substring(currentPowerPlayIndex,

currentPowerPlayIndex + 2));

//TODO

}

}