**CSC 20 – Exam 1 Preparation**

This page is a guide for getting ready for the module exam. You should read all of the readings and work through all of the practice assignments. This is too much work to do at the last minute, so start early and get help if you need it. If you have time, you are encouraged to do additional similar CodeStepByStep problems for more practice. You may work alone or in groups.

**String class**

Objective: Student should be able to use methods in the String class to access parts of Strings and query its contents.

Read/Study: "String Objects" subsection of BJP 3.3.

Practice methods (you may see methods of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/strings/printBackward>  
<https://www.codestepbystep.com/problem/view/java/strings/nameDiamond>

Practice programs (you may see programs of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/strings/NameGame>

Practice problems (you may see problems like this on exams):

<https://www.codestepbystep.com/problem/view/java/strings/marshallMathers>  
<https://www.codestepbystep.com/problem/view/java/strings/dracula>

**Random class**

Objective: Student should be able to use methods in the Random class to generate uniformly distributed numbers over arbitrary ranges.

Read/Study: "Random Numbers" subsection of BJP 5.1.

Practice methods (you may see methods of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/parameters/randomOver>  
<https://www.codestepbystep.com/problem/view/java/parameters/coinFlip>  
<https://www.codestepbystep.com/problem/view/java/parameters/randomWalk>

Practice programs (you may see programs of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/loops/RollTwoDice>  
<https://www.codestepbystep.com/problem/view/java/loops/Piglet>

Practice problems (you may see problems like this on exams):

Check “Exam 1 Practice Problems”

**Scanner class**

Objective: Student should be able to create Scanner object initialized with String, File, or standard-input. Student should be able to use methods in the Scanner class to test whether a token type or line is available and read it.

Read/Study: "Interactive Programs and Scanner Objects" and "Sample Interactive Program" subsections of BJP 3.3. BJP 5.4. BJP 6.2 through and including subsection "Scanner Parameters".

Practice methods (you may see methods of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/fileio/removeDuplicatesFromFile>  
<https://www.codestepbystep.com/problem/view/java/fileio/flipCoins>

Practice programs (you may see programs of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/loops/FizzBuzz>  
<https://www.codestepbystep.com/problem/view/java/loops/SentinelSum>

Practice problems (you may see problems like this on exams):

Check “Exam 1 Practice Problems”

**Arrays**

Objective: Student should be able to create 1D and 2D arrays and write loops that read or write to them. Student should be able to explain "reference semantics".

Read/Study: BJP 7.1, 7.2, 7.3. BJP 7.5 through and including subsection "Rectangular Two-Dimensional Arrays".

Practice methods (you may see methods of this difficulty on exams):

<https://www.codestepbystep.com/problem/view/java/arrays/print>  
<https://www.codestepbystep.com/problem/view/java/arrays/maxValue>  
<https://www.codestepbystep.com/problem/view/java/arrays/sorted>  
<https://www.codestepbystep.com/problem/view/java/arrays/2d/matrixSum>

Practice problems (you may see problems like this on exams):

Check “Exam 1 Practice Problems”

<https://www.codestepbystep.com/problem/view/java/arrays/squaredArray>  
<https://www.codestepbystep.com/problem/view/java/arrays/mystery1>  
<https://www.codestepbystep.com/problem/view/java/arrays/2d/array2dMystery>

**Exceptions**

Objective: Student should be able to throw exceptions and use try/catch to handle exceptions.

Read/Study: BJP 4.4 through and including subsection "Throwing Exceptions". "The try/catch Statement" in BJP Appendix C.

Practice problems (you may see problems like this on exams):

Check “Exam 1 Practice Problems”

**Formatting conventions**

Objective: Student should be able to follow standard formatting conventions when writing Java code.

Familiarize yourself with: <https://google.github.io/styleguide/javaguide.html>

Practice problems (you may see problems like this on exams):

Check “Exam 1 Practice Problems”