

# Learning Guide Unit 1

Site: [University of the People](#)  
Course: CS 1103-01 Programming 2 - AY2023-T1  
Book: Learning Guide Unit 1

Printed by: Abel Lifaefi Mbula  
Date: Saturday, 29 October 2022, 1:59 AM

## Description

Learning Guide Unit 1

## Table of contents

**Overview**

**Introduction**

**Reading Assignment**

**Discussion Forum Question**

**Assignment**

**Learning Journal Assignment**

**Exercises**

# Overview

---

## Unit 1: Exceptions, the switch statement, Analysis of Algorithms

---

### Topics:

This unit will address the following topics:

- A review of some topics in the CS 1102 course
- The switch statement
- Advanced topics in exceptions; the try..catch statement
- Analysis of algorithms

---

### Learning Objectives:

- Practically review the features of Eclipse
- Refresh your knowledge of top-down design, bottom-up design, pseudocode, and modularity
- Review techniques for creating arrays, initializing arrays, and processing arrays.
- Study the following:
  - Exceptions and exception classes
  - The try statement
  - Throwing exceptions
  - Mandatory exception handling
  - Programming with exceptions
- Learn some techniques and results of Analysis of Algorithms.

---

### Tasks:

- Read the assigned material in the textbook.
- Try to do as many of the exercises in Chapter 8 of the textbook as possible in order to reinforce your knowledge.
- Also, do the non-graded exercise that appears at the end of this Learning Guide and post your solutions to the Learning Journal.
- Perform Labs 1 and 2.
- Post your answer to the question in the Discussion Forum and follow it up with approximately 3-4 comments to other students' posts. Rate other students' posts.
- Submit your Assignment.
- Post your activities throughout the week in the Learning Journal.
- Test yourself by taking the self-quiz.

# Introduction

---

## Welcome back! There are two parts to this week's study program:

1. The first part will give you a chance to get back into the programming groove. We will review some of the basic ideas of program design, such as top-down design, bottom-up design, pseudocode, and modularity. We will briefly cover some basic features of Java that you might not have paid much attention to in CS 1102, namely the switch statement (Section 8.2.10), JavaDoc (Section 4.5.4), and (in more detail) exceptions (Section 8.3). Lab 1 will introduce Eclipse, the programming environment that you will be using for all of the programs that you write.

You are requested to read -- and pay attention to -- a style guide for programming in Java. The style guide can be found permanently [here](#)

2. The second half of the week you should spend on new material: The reading for this part is Section 8.6, which is an introduction to the analysis of algorithms. Part of Lab 2 is a demonstration of the fact that two algorithms that accomplish the same task can have very different run times, which indicates that it can be worthwhile to analyze the running time of the algorithms that we write and to search for more efficient algorithms.

The second part of the lab involves writing a lab that uses exceptions and the try..catch statement.

### switch

The switch statement is used far less often than the if statement, but it is sometimes useful for expressing a certain type of multi-way branch.

A switch statement allows you to test the value of an expression and, depending on that value, to jump directly to some location within the switch statement. Only expressions of certain types can be used. The positions that you can jump to are marked with case labels that take the form: "case constant:". As the final case in a switch statement you can, optionally, use the label "default:".

### Exceptions

The term exception is shorthand for the phrase "exceptional event." An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions.

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an *exception object*, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called *throwing an exception*.

After a method throws an exception, the runtime system attempts to find something to handle it. The set of possible mechanisms to handle the exception is the ordered list of subroutines that had been called to get to the method where the error occurred. The list of subroutines is known as the call stack.

### Analysis of Algorithms

This is a field of computer science dedicated to analyzing the efficiency of programs.

One of the main techniques of analysis of algorithms is **asymptotic analysis**. The term "asymptotic" here means basically "the tendency in the long run." An asymptotic analysis of an algorithm's run time looks at the question of how the run time depends on the size of the problem.

Section 8.6 provides an informal introduction to this discipline -- even though it is a theoretical field, every working programmer should be aware of some of its techniques and results.



## Reading Assignment

Eck, D. J. (2019). *Introduction to programming using Java, version 8.1*. Hobart and William Smith Colleges. <http://math.hws.edu/javanotes>

For this week, you should read the following material (some of it is a review of topics that were covered in the CS 1102 course):

- **Section 3.6.1: The Basic switch Statement**
- **Section 4.6.5: Javadoc (review)**
- **Section 7.2: Array Processing (review)**
- **Section 7.3: ArrayLists (review)**
- **Section 7.4: Searching and Sorting (review)**
- **Section 7.5: Two-dimensional Arrays (review)**
- **Section 8.3: Exceptions and Try..catch**
- **Section 8.5: Analysis of Algorithms**

### Video Resources

Codecourse. (2014, June 13). *Java for beginners: Switch statements & enumerations (6/10)* [Video]. YouTube.

Java For Beginners: Switch Statements...



Joe James. (2015, January 16). *Java: Array vs ArrayList difference* [Video]. YouTube.

Java: Array vs ArrayList Difference



## Discussion Forum Question

Who should be responsible for catching and handling exceptions: the computer or the computer program?



# Assignment

## Assignment

---

This week's assignment is Part 1: Benchmarking Sorting Algorithms from Lab 2 of this unit.

## Learning Journal Assignment

Your learning journal entry must be a reflective statement that considers the following questions:

- Describe what you did. This does not mean that you copy and paste from what you have posted or the assignments you have prepared. You need to describe what you did and how you did it.
- Describe your reactions to what you did
- Describe any feedback you received or any specific interactions you had. Discuss how they were helpful
- Describe your feelings and attitudes
- Describe what you learned

Another set of questions to consider in your learning journal statement include:

- What surprised me or caused me to wonder?
- What happened that felt particularly challenging? Why was it challenging to me?
- What skills and knowledge do I recognize that I am gaining?
- What am I realizing about myself as a learner?
- In what ways am I able to apply the ideas and concepts gained to my own experience?

Your Learning Journal must be a minimum of 500 words.

## Exercises

This is a non-graded exercise which should be posted to your learning journal.

Write a program that uses the following subroutine, from Subsection 8.3.3 of the textbook, to solve equations specified by the user.

```
/**
 * Returns the larger of the two roots of the quadratic equation
 *  $Ax^2 + Bx + C = 0$ , provided it has any roots. If  $A == 0$  or
 * if the discriminant,  $B^2 - 4AC$ , is negative, then an exception
 * of type IllegalArgumentException is thrown.
 */
static public double root( double A, double B, double C )
    throws IllegalArgumentException {
    if (A == 0) {
        throw new IllegalArgumentException("A can't be zero.");
    }
    else {
        double disc = B*B - 4*A*C;
        if (disc < 0)
            throw new IllegalArgumentException("Discriminant < zero.");
        return (-B + Math.sqrt(disc)) / (2*A);
    }
}
```

Your program should allow the user to specify values for A, B, and C. It should call the subroutine to compute a solution of the equation. If no error occurs, it should print the root. However, if an error occurs, your program should catch that error and print an error message. After processing one equation, the program should ask whether the user wants to enter another equation. The program should continue until the user answers no.