SOFTWARE

JAVA, ALGORITHMS & DATA TYPES

REVIEWING SOFTWARE

PROGRAMS, APPLICATIONS, AND APPS

INTRODUCTION

- Briefly introduce the topic and its relevance in programming.
- Mention that these terms are often used interchangeably but have distinct meanings.
- Highlight the importance of understanding these concepts for effective software development.

WHAT IS SOFTWARE?

software - Collection of programs, data, and instructions managing computer hardware.

- Includes operating systems, utilities, libraries, applications, etc.
- Broad term covering all aspects of computer-related programs and data.

WHAT IS A COMPUTER PROGRAM?

computer program – a sequence of instructions in a programming language.

- Purpose: Guides the computer in performing specific tasks.
- Basic building blocks of software development.

WHAT IS AN APPLICATION?

<u>application</u> - user-facing software designed for specific functions.

- Can be general-purpose or specialized.
- Examples: Word processors, media players, productivity tools.

WHAT IS AN APP?

app - subset of applications designed for mobile devices.

- Term derived from "application."
- Installed from app stores; optimized for touch-based interfaces.

KEY DIFFERENCES

- Scope and Purpose
- Software: Broad term for all computer-related programs and data.
- Computer Program: Specific set of instructions for a task.
- Application: User-oriented software with specific functions.
- App: Mobile application designed for mobile devices.

KEY DIFFERENCES (CONT'D)

- Platforms
- Software: Runs on various platforms (desktop, servers, embedded systems).
- Computer Program: Not platform-limited.
- Application: Platform-dependent (desktop and mobile).
- App: Specifically designed for mobile platforms.

KEY DIFFERENCES (CONT'D)

- Mobile vs. Desktop:
- Apps: Associated with mobile devices.
- Applications: Run on both desktop and mobile platforms.

KEY DIFFERENCES (CONT'D)

- Usage:
- Software and Computer Program: Technical terms for programming discussions.
- Application and App: User-centric terms for software discussions.

HOW IS A PROGRAM EXECUTED?

Interpreting

- Definition: Process of executing source code directly by an interpreter
- Interpreter reads and executes code line-by-line
- Slower execution compared to compilation

Compiling

- Definition: Process of converting source code into machine code or bytecode
- Compiler translates entire code before execution
- Faster execution compared to interpretation

INTERPRETING VS. COMPILING

Interpreting

- Code is executed line-by-line.
- Immediate error detection.
- Slower execution.
- Examples: Python, Ruby

Compiling

- Code translated before execution.
- Optimized performance.
- Potential delay in error detection.
- Examples: C++, Java

JAVA PROGRAMMING LANGUAGE

Can be used to develop

- standalone applications.
- applications running from a browser.
- applications for hand-held devices.
- applications for Web servers.

JAVA'S HISTORY

- James Gosling and Sun Microsystems
- Oak
- Java, May 20, 1995, Sun World
- HotJava
  - The first Java-enabled Web browser

- JDK 1.02 (1995)
- JDK 1.1 (1996)
- JDK 1.2 (1998)
- JDK 1.3 (2000)
- JDK 1.4 (2002)
- JDK 1.5 (2004) a. k. a. JDK 5 or Java 5
- JDK 1.6 (2006) a. k. a. JDK 6 or Java 6
- JDK 1.7 (2011) a. k. a. JDK 7 or Java 7
- JDK 1.8 (2014) a. k. a. JDK 8 or Java 8
- Java 9, 10, 11, 12, 13, 14

https://en.wikipedia.org/wiki/Java_version_history

THE JAVA LANGUAGE SPECIFICATION

- The *Java language specification* is a technical definition of the Java programming language's syntax and semantics
- The *application program interface (API)*, also known as library, contains predefined classes and interfaces for developing Java programs

JDK EDITIONS

- Java SE (Standard Edition):
  - Core libraries, tools for desktop and server applications.
  - Widely used for general-purpose programming.

- Java EE (Enterprise Edition):
  - Extensions for large-scale enterprise applications.
  - Components for distributed computing, web services, etc.
- Java ME (Micro Edition):
  - Platform for mobile and embedded systems.
  - Suitable for resource-constrained devices

CREATING, COMPILING, AND EXECUTING A JAVA PROGRAM

- A Java compiler translates a Java source file into a Java bytecode file
- The compiler generates a bytecode file with a `.class` extension
- Bytecode is architecture neutral and can run on any platform that has a Java Virtual Machine (JVM)
- Java source code is ***compiled*** into Java bytecode and Java bytecode is ***interpreted*** by the JVM
  - Individual instructions in the bytecode are translated into the target machine language code one at a time
- The JVM executes your code along with the code in the library

JAVA VIRTUAL MACHINE (JVM)

- Executes compiled Java bytecode
- Platform-independent execution environment
- Converts bytecode to native machine code
- Ensures Java's "write once, run anywhere" capability

JAVA RUNTIME ENVIRONMENT (JRE)

- Provides runtime environment for Java applications
- Includes JVM, libraries, and other resources.
- Necessary to run compiled Java programs

JVM VS. JRE

- JVM
  - Converts bytecode to native code
  - Part of the JRE
  - Executes Java programs

- JRE
  - Includes JVM, libraries, and resources
  - Required to run Java applications

JAVA DEVELOPMENT KIT (JDK)

- Definition: Software package for developing Java applications.
- Includes JRE, compiler, debugger, and other tools.
- Versions
  - Standard Edition (SE)
  - Enterprise Edition (EE)
  - Micro Edition (ME).

MORE JAVA VERSION INFO

- Guide to Java Versions and Features
- What is the JDK? Introduction to the Java Development Kit
- What is the JVM? Introducing the Java virtual machine
- What is the JRE? Introduction to the Java Runtime Environment
- OpenJDK

OpenJDK

- OpenJDK (Open Java Development Kit) is a free and open-source implementation of Java SE
- Licensed under the GPL-2.0-only
- The official reference implementation of Java SE since version 7
- Due to Oracle no longer releasing updates for long-term support (LTS) releases under a permissive license, others have begun offering implementations

JAVA IMPLEMENTATIONS

Oracle/OpenJDK

- Oracle's official implementation.
- OpenJDK is the open-source version.

Adoptium / Eclipse Temurin

- Community-driven distribution.
- Focus on providing timely updates and long-term support.
- formerly AdoptOpenJDK

Amazon Corretto

- Amazon's free distribution of OpenJDK.
- Optimized for production environments.

Zulu by Azul Systems

- Open-source OpenJDK distribution.
- Emphasis on performance and stability.

VISUAL STUDIO CODE

ALGORITHMS

WHAT IS AN ALGORITHM?

algorithm – a sequence of instructions that solves a problem or achieves a result

- The term comes from the polymath Muhammad ibn Musa al-Khwarizmi
  - described as the father of algebra
- algorithm is Al-Khwarizmi translated into Latin

- Form the basis of computer programming
- First computer algorithm created by Ada Lovelace
- Make it possible for computers to perform tasks that are challenging or impossible for humans

Monumento a Muhammad al-Juarismi en la Ciudad Universitaria de Madrid

ALGORITHM CHARACTERISTICS

- **Clear and Unambiguous**: Each of its steps should be clear in all aspects and must lead to only one meaning
- **Well-Defined Inputs**: It may or may not take input.
- **Well-Defined Outputs:** produce at least 1 output.
- **Finite-ness:** The algorithm must be finite, i.e. it should terminate after a finite time.
- **Feasible:** The algorithm must be simple, generic, and practical, such that it can be executed with the available resources.
- **Language Independent**
- **Definiteness:** instructions must be unambiguous, precise, and easy to interpret; clearly understand what is to be done
- **Finiteness:** must terminate after a finite number of steps in all test cases

ADVANTAGES & DISADVANTAGES

- Easy to understand
- A step-wise representation of a solution to the problem
- The problem is broken down into smaller pieces or steps

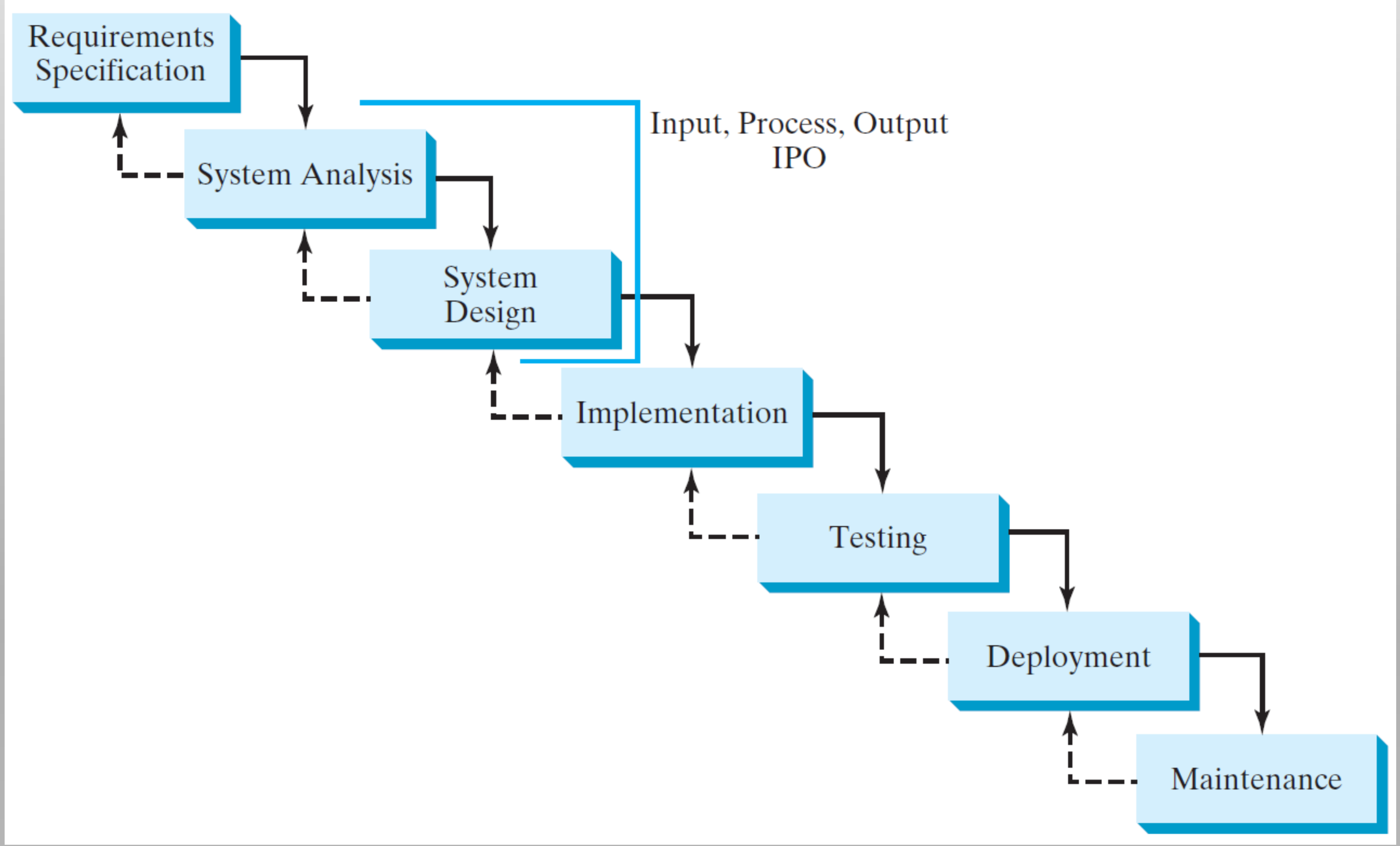
- Can be time-consuming
- Understanding complex logic through algorithms can be very difficult
- Branching and Looping statements are difficult to illustrate

HOW TO CREATE AN ALGORITHM

Identify the IPO

- Input
- Process
- Output
- A rudimentary software development methodology

SOFTWARE DEVELOPMENT PROCESS



REQUIREMENTS SPECIFICATION

- A formal process that seeks to understand the problem that the software will address and to document in detail what the software system needs to do
- Involves close interaction between users and developers.
- Developers need to study the problem carefully to identify what the software needs to do

SYSTEM ANALYSIS

- Seeks to analyze the data flow and to identify the system's input and output
- When you do analysis, it helps to identify what the output is first, and then figure out what input data you need in order to produce the output

SYSTEM DESIGN

- Design a process for obtaining the output from the input
- Involves the use of many levels of **abstraction** to break down the problem into manageable components and design strategies for implementing each component.
- View each component as a subsystem that performs a specific function of the system
- The essence of system analysis and design is **input, process, and output (IPO)**

COMPUTATIONAL THINKING

computational thinking - a problem-solving approach used in computer science and other fields that involves breaking down complex problems into smaller, manageable parts, and then solving them

Decomposition

- Breaking a problem into smaller, more manageable sub-problems
- Makes it easier to understand and tackle complex issues

Pattern Recognition

- Identifying patterns or similarities among different problems
- Using these patterns to create general solutions that can be applied to various situations

Abstraction

- Focusing on the important details; ignoring unnecessary complexities.
- Allows you to create higher-level representations that capture the essential features of a problem

Algorithmic Design

- Creating step-by-step instructions that perform the necessary actions to solve a problem

IDENTIFYING THE BETTER ALGORITHM

- Some algorithms are better than others
- Suppose two algorithms perform the same task, how can we identify the better one?

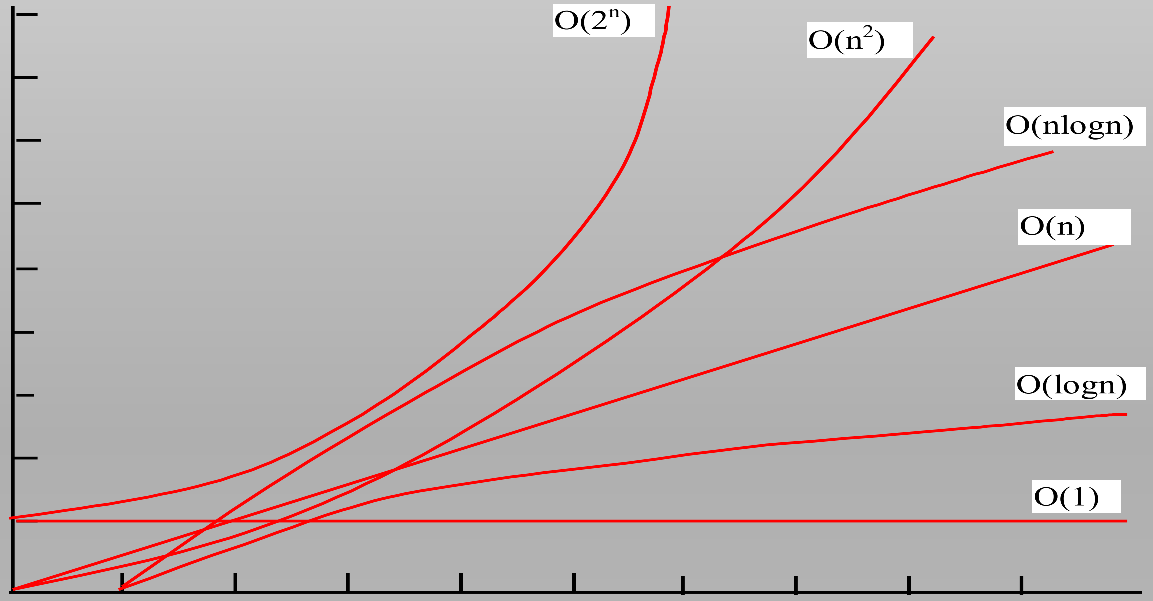
APPROACHES TO DETERMINING THE BETTER ALGORITHM

Approaches

- Execution time
  - Various processes running concurrently ➢ system load dependent
  - Input dependent
- Growth rate
  - analyze algorithms independent of computers and specific input
  - approximates the effect of a change on the size of the input
  - Observe the rate an algorithm's execution time increases as the input size increases
  - So, compare two algorithms by examining their growth rates

Big O Notation

- The growth rate has an order of magnitude of n.
- Computer scientists use the Big O notation to abbreviate for "order of magnitude."



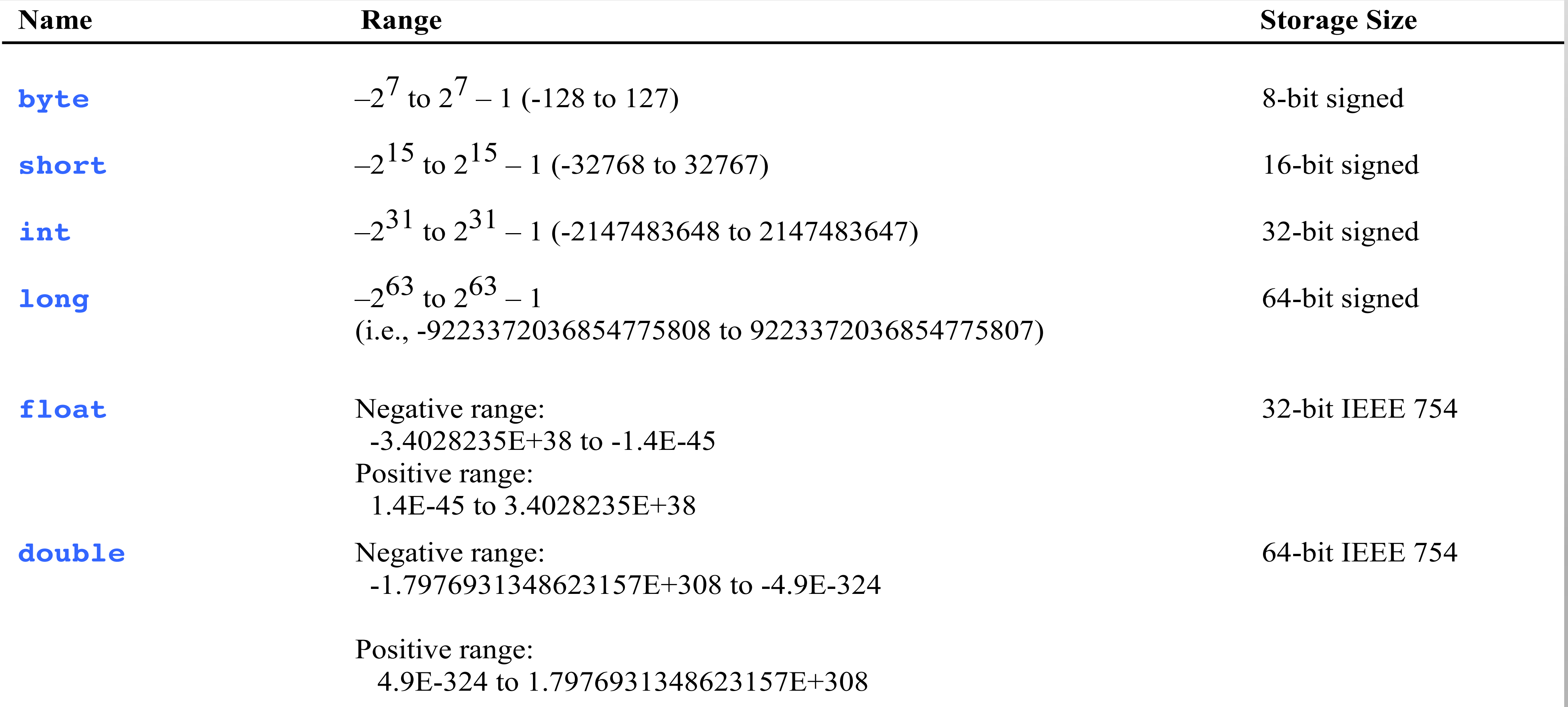
THE ALGORITHM AGE

- How Big Data Is Growing
- *"…experts worry they can also put too much control in the hands of corporations and governments, perpetuate bias, create filter bubbles, cut choices, creativity and serendipity, and could result in greater unemployment..."* (Rainie, 2022)
- *The internet, smartphones, GPS mapping systems, what we see on social media, hacking, cyberattacks*
- The use of algorithms is spreading as massive amounts of data are being created, captured and analyzed by businesses and governments. Some are calling this the Age of Algorithms

- *The British pound dropped 6.1% in value in seconds on Oct. 7, 2016, partly because of currency trades triggered by algorithms.*
- Microsoft engineers created a Twitter bot named "Tay" this past spring in an attempt to chat with Millennials by responding to their prompts, but within hours it was spouting racist, sexist, Holocaust-denying tweets based on algorithms that had it "learning" how to respond to others based on what was tweeted at it.
- Facebook tried to create a feature to highlight Trending Topics from around the site in people's feeds. First, it had a team of humans edit the feature, but controversy erupted when some accused the platform of being biased against conservatives. So, Facebook then turned the job over to algorithms only to find that they could not discern real news from fake news.

DATA TYPES

NUMERICAL DATA TYPES



Name	Range	Storage Size
`byte`	-2^{7} to $2^{7} - 1$ (-128 to 127)	8-bit signed
`short`	-2^{15} to $2^{15} - 1$ (-32768 to 32767)	16-bit signed
`int`	-2^{31} to $2^{31} - 1$ (-2147483648 to 2147483647)	32-bit signed
`long`	-2^{63} to $2^{63} - 1$ (i.e., -9223372036854775808 to 9223372036854775807)	64-bit signed
`float`	Negative range: -3.4028235E+38 to -1.4E-45 Positive range: 1.4E-45 to 3.4028235E+38	32-bit IEEE 754
`double`	Negative range: -1.7976931348623157E+308 to -4.9E-324 Positive range: 4.9E-324 to 1.7976931348623157E+308	64-bit IEEE 754

NUMERIC OPERATORS

Name	Meaning	Example	Result
+	Addition	34 + 1	35
-	Subtraction	34.0 – 0.1	33.9
*	Multiplication	300 * 30	9000
/	Division	1.0 / 2.0	0.5
%	Remainder	20 % 3	2

CHARACTER DATA TYPE

Four hexadecimal digits.

```
char letter = 'A'; (ASCII)
char numChar = '4'; (ASCII)
char letter = '\u0041'; (Unicode)
char numChar = '\u0034'; (Unicode)
```

REFERENCES

Liang, Y. D. (2022). *Introduction to java programming and Data Structures: Comprehensive Version*. Pearson Education Limited.

OpenJDK. (n.d.). https://openjdk.org/

Colebourne, S. (n.d.). Time to look beyond Oracle's JDK. https://blog.joda.org/2018/09/time-to-look-beyond-oracles-jdk.html

GeeksforGeeks. (2023, August 3). *What is algorithm: Introduction to algorithms*. GeeksforGeeks. https://www.geeksforgeeks.org/introduction-to-algorithms/

Rainie, L. (2022, September 15). *Code-dependent: Pros and cons of the algorithm age*. Pew Research Center: Internet, Science & Tech. https://www.pewresearch.org/internet/2017/02/08/code-dependent-pros-and-cons-of-the-algorithm-age/

Alang, N. (2016, May 13). *Life in the age of algorithms*. The New Republic. https://newrepublic.com/article/133472/life-age-algorithms