**Chapter 1**

**Self-Review Exercises 1.1**

**Fill in the blanks in each of the following statements:**

a) Computers process data under the control of sets of instructions called computer programs.

b) The key logical units of the computer are the **Input unit** , **Output unit**, **Memory unit**, **Arithmetic and logic unit(ALU)**, **Central processing unit (CPU**) and **Secondary storage unit**.

c) The three types of languages they are **machine language, assembly language** and High-level Language.

d) The programs that translate high-level language programs into machine language are called **compilers**.

e) **Android** is an operating system for mobile devices based on the Linux kernel and Java.

f) **Release candidates** software is generally feature complete, (supposedly) bug free and ready for use by the community.

g) The Wii Remote, as well as many smartphones, use a(n) **accelerometer** which allows the device to respond to motion.

**1.2 Fill in the blanks in each of the following sentences about the Java environment:**

a) The **Java** command from the JDK executes a Java application.

b) The **Javac** command from the JDK compiles a Java program.

c) A Java source code file must end with the .**java** file extension.

d) When a Java program is compiled, the file produced by the compiler ends with the **.class** file extension.

e) The file produced by the Java compiler contains **bytecodes** that are executed by the Java Virtual Machine.

**1.3 Fill in the blanks in each of the following statements**

a) Objects enable the design practice of **information hiding** —although they may know how to communicate with one another across well-defined interfaces, they normally are not allowed to know how other objects are implemented.

b) Java programmers concentrate on creating **classes**, which contain fields and the set of methods that manipulate those fields and provide services to clients.

c) The process of analyzing and designing a system from an object-oriented point of view is called **object-oriented analysis and design (OOAD)**.

d) A new class of objects can be created conveniently by **inheritance** —the new class (called the subclass) starts with the characteristics of an existing class (called the superclass), possibly customizing them and adding unique characteristics of its own.

e) **The Unified Modeling Language (UML)** is a graphical language that allows people who design software systems to use an industry-standard notation to represent them.

f) The size, shape, color and weight of an object are considered **attributes** of the object’s class.

**Exercises 1.4 Fill in the blanks in each of the following statements:**

a) The logical unit that receives information from outside the computer for use by the computer is the **input unit**.

b) The process of instructing the computer to solve a problem is called **programming** .

c) **Assembly language** is a type of computer language that uses English-like abbreviations for machine-language instructions.

d) **Output unit** is a logical unit that sends information which has already been processed by the computer to various devices so that it may be used outside the computer.

e) **Secondary storage unit** and primary **memory** are logical units of the computer that retain information.

f) **Arithmetic and logic unit (ALU)** is a logical unit of the computer that performs calculations.

g) **Arithmetic Logic Unit (ALU)** is a logical unit of the computer that makes logical decisions.

h) **High-level languages** are most convenient to the programmer for writing programs quickly and easily.

i) The only language a computer can directly understand is that computer’s **machine language**.

j) **Central processing unit (CPU)** is a logical unit of the computer that coordinates the activities of all the other logical units.

**1.5 Fill in the blanks in each of the following statements:**

a) The **Java** programming language is now used to develop large-scale enterprise applications, to enhance the functionality of web servers, to provide applications for consumer devices and for many other purposes.

b) **C** initially became widely known as the development language of the UNIX operating system.

c) The **Transmission Control Protocol (TCP)** ensures that messages, consisting of sequentially numbered pieces called bytes, were properly routed from sender to receiver, arrived intact and were assembled in the correct order.

d) The **C++** programming language was developed by Bjarne Stroustrup in the early 1980s at Bell Laboratories.

1.6 Fill in the blanks in each of the following statements:

a) Java programs normally go through five phases— **edit**, **compile**, **load**, **verify**, and **execute**.

b) A(n) **Integrated Development Environment (IDE)** provides many tools that support the software development process, such as editors for writing and editing programs, debuggers for locating logic errors in programs, and many other features.

c) The command **java invokes the Java Virtual Machine (JVM),** which executes Java programs.

d) A(n) **virtual machine** is a software application that simulates a computer, but hides the underlying operating system and hardware from the programs that interact with it.

e) The **class loader** takes the .class files containing the program’s bytecodes and transfers them to primary memory. f) The examines bytecodes to ensure that they’re valid.

f) The **bytecode verifier** examines bytecodes to ensure that they’re valid.

1.7 Explain the two compilation phases of Java programs.

***ANSWER***

**1. Compilation Phase (Source Code → Bytecode)**

* **Input: .java source file (written by the programmer).**
* **Process: The Java compiler (javac) translates the human-readable Java code into bytecode.**
* **Output: .class file containing bytecode.**

**Key points:**

* **Bytecode is platform-independent, meaning it can run on any machine that has a Java Virtual Machine (JVM).**
* **The compiler checks for syntax errors during this phase. If there are errors, it stops here.**
* **This phase does not execute the program; it only prepares it for execution.**

**E.G**

**javac MyProgram.java**

**2. Execution Phase (Bytecode → Machine Code)**

* **Input: .class bytecode file.**
* **Process: The Java Virtual Machine (JVM) reads the bytecode and interprets it (or uses Just-In-Time compilation to convert it into native machine code).**
* **Output: The program runs on the computer, performing the tasks defined in the code.**

**Key points:**

* **This phase makes Java programs platform-independent because the JVM handles the translation to the machine-specific instructions.**
* **The JVM also performs runtime checks, like verifying bytecode safety and managing memory (garbage collection).**

**E.G**

**java MyProgram**

1.8 One of the world’s most common objects is a wrist watch. Discuss how each of the following terms and concepts applies to the notion of a watch: object, attributes, behaviors, class, inheritance (consider, for example, an alarm clock), modeling, messages, encapsulation, interface and information hiding.

**Making a Difference**

***ANSWER***

**1. Object**

**An object is a specific, real-world instance of something — in this case, a particular wristwatch.**

* **Example: *Your personal wristwatch* is an object.**
* **It has its own state (like current time, brand, or battery level).**

**2. Attributes (or Properties)**

**Attributes describe the characteristics or data of an object.  
For a wristwatch, attributes might include:**

* **Color (silver, black, gold)**
* **Brand (Rolex, Casio, Apple)**
* **Time (hours, minutes, seconds)**
* **Date**
* **Battery level**
* **Material (leather strap, metal strap)**

**These are the data fields that define the state of the watch object.**

**3. Behaviors (or Methods)**

**Behaviors describe what the object can do — its actions or functions.  
A wristwatch’s behaviors could include:**

* **Display time**
* **Set time**
* **Show date**
* **Start or stop stopwatch**
* **Ring alarm (if digital or smart)**

**Each behavior is like a method in programming.**

**4. Class**

**A class is like a blueprint or template from which objects are created.**

* **The class WristWatch defines what attributes and behaviors any watch should have.**
* **Example:**
* **class WristWatch {**
* **String brand;**
* **String color;**
* **void showTime() { ... }**
* **void setTime() { ... }**
* **}**

**Every real watch object (Rolex, Casio, etc.) is created from this class.**

**5. Inheritance**

**Inheritance allows a class to derive from another class and reuse or extend its features.**

* **Example: A SmartWatch or AlarmClock could inherit from the WristWatch class.**
* **class AlarmClock extends WristWatch {**
* **void setAlarm() { ... }**
* **}**

**The alarm clock inherits all features of a wristwatch and adds new ones (like setAlarm() or ringAlarm()).**

**6. Modeling**

**Modeling means representing real-world things (like a watch) in a program using classes and objects.  
In this case, we model the real wristwatch as a software object with its data (attributes) and actions (behaviors).**

**7. Messages**

**Objects communicate with each other by sending messages, usually in the form of method calls.**

* **Example: When the user presses a button to set the time, a message like watch.setTime() is sent to the watch object.**
* **In OOP, messages = requests sent to an object to perform an operation.**

**8. Encapsulation**

**Encapsulation means bundling data (attributes) and methods (behaviors) inside a single unit (the object).**

* **The wristwatch’s internal working (time calculation, ticking mechanism) is hidden from the user.**
* **The user only interacts through the buttons or screen, not by directly changing the internal time variables.**

**9. Interface**

**An interface defines how other objects or users interact with the watch — what actions are available without revealing how they work.**

* **Example: Buttons or touchscreen controls are the watch’s interface.**
* **In code, it could be something like:**
* **interface TimeDevice {**
* **void showTime();**
* **void setTime();**
* **}**

**A WristWatch or SmartWatch class can implement this interface differently.**

**10. Information Hiding**

**Information hiding means keeping the internal details private and exposing only what’s necessary.**

* **For a watch, the user doesn’t need to know how the internal clock mechanism works — they just see the time.**
* **In programming:**
* **private int batteryLevel;**
* **public void showBatteryLevel() { ... }**

**The internal data (batteryLevel) is hidden, and only specific methods are public.**

***1.9 (Test-Drive: Carbon Footprint Calculator)***

Some scientists believe that carbon emissions, especially from the burning of fossil fuels, contribute significantly to global warming and that this can be combatted if individuals take steps to limit their use of carbon-based fuels. Organizations and individuals are increasingly concerned about their “carbon footprints.” Websites such as TerraPass http://www.terrapass.com/carbon-footprint-calculator/ and Carbon Footprint http://www.carbonfootprint.com/calculator.aspx provide carbon-footprint calculators. Test-drive these calculators to determine your carbon footprint. Exercises in later chapters will ask you to program your own carbon-footprint calculator. To prepare for this, use the web to research the formulas for calculating carbon footprints.

***ANSWER***

**1. Carbon Footprint from Driving**

**When you drive, your car burns fuel (petrol or diesel), which releases carbon dioxide (CO₂).**

**🧮 Formula:**

**⚙️ Emission factors:**

* **Petrol (gasoline): 2.31 kg CO₂ per liter**
* **Diesel: 2.68 kg CO₂ per liter**

**✅ Example:**

**If you use 100 liters of petrol in a month:**

**In a year:**

**💡 2. Carbon Footprint from Electricity Use**

**Electricity generation often burns coal or gas, producing CO₂.  
(The amount depends on how “green” your country’s energy is.)**

**🧮 Formula:**

**⚙️ Emission factor (average):**

* **0.5 kg CO₂ per kWh (varies by country)**

**✅ Example:**

**If your home uses 300 kWh of electricity per month:**

**In a year:**

**✈️ 3. Carbon Footprint from Air Travel**

**Airplanes release a lot of CO₂ at high altitudes.**

**🧮 Formula:**

**⚙️ Emission factor (economy class):**

* **0.09 kg CO₂ per km per passenger**

**✅ Example:**

**A 2,000 km flight (about 3 hours):**

**🌍 4. Total Carbon Footprint (Example)**

| **Activity** | **CO₂ per year** |
| --- | --- |
| **Driving** | **2.8 tons** |
| **Electricity** | **1.8 tons** |
| **Flying** | **0.2 tons** |
| **Total** | **≈ 4.8 tons of CO₂/year** |

***1.10 (Test-Drive: Body Mass Index Calculator)***

Obesity causes significant increases in illnesses such as diabetes and heart disease. To determine whether a person is overweight or obese, you can use a measure called the body mass index (BMI). The United States Department of Health and Human Services provides a BMI calculator at http://www.nhlbi.nih.gov/guidelines/obesity/BMI/ bmicalc.htm. Use it to calculate your own BMI. A forthcoming exercise will ask you to program your own BMI calculator. To prepare for this, use the web to research the formulas for calculating BMI.

**ANSWER**

**⚙️ BMI Formula**

**There are two common formulas, depending on whether you use metric or imperial units:**

**🔹 Metric System (kg and meters):**

**Example:  
If someone weighs 68 kg and is 1.65 m tall:**

**✅ Result: BMI = 24.98 → Normal weight**

**🔹 Imperial System (pounds and inches):**

**Example:  
If someone weighs 150 lb and is 65 inches tall:**

**✅ Result: BMI = 24.96 → Normal weight**

**📊 BMI Categories (According to WHO/CDC)**

| **BMI Value** | **Category** |
| --- | --- |
| **Less than 18.5** | **Underweight** |
| **18.5 – 24.9** | **Normal weight** |
| **25 – 29.9** | **Overweight** |
| **30 and above** | **Obese** |

***1.11 (Attributes of Hybrid Vehicles)***

Hybrid vehicles are becoming increasingly popular, because they often get much better mileage than purely gasoline-powered vehicles. Browse the web and study the features of four or five of today’s popular hybrid cars, then list as many of their hybrid-related attributes as you can. Some common attributes include city-miles-per-gallon and highway-miles-per-gallon. Also list the attributes of the batteries (type, weight, etc.).

***ANSWER***

**1) Toyota Prius (2025)**

* **Fuel economy: ~57 mpg combined.** [**Car and Driver+2CarBuzz+2**](https://www.caranddriver.com/toyota/prius?utm_source=chatgpt.com)
* **Powertrain: 2.0-litre inline-4 + electric motors (front-wheel drive), ~194 hp.** [**Car and Driver+1**](https://www.caranddriver.com/toyota/prius?utm_source=chatgpt.com)
* **Battery pack: Lithium-ion (for many recent trims).** [**Le Guide de l'auto+1**](https://www.guideautoweb.com/en/makes/toyota/prius/2025/specifications/xle/?utm_source=chatgpt.com)
* **Plug-in version specs: Battery capacity ~13.6 kWh, curb weight ~3,461 lb for plug-in version.** [**Kbb.com+1**](https://www.kbb.com/toyota/prius-plug-in-hybrid/2025/specs/?utm_source=chatgpt.com)
* **Vehicle curb weight: ~3,461 lb (plug-in version)** [**Kbb.com**](https://www.kbb.com/toyota/prius-plug-in-hybrid/2025/specs/?utm_source=chatgpt.com)

**2) Toyota Corolla Hybrid (2025)**

* **Battery pack: Type = Lithium-ion.** [**toyotacanada.scene7.com+1**](https://toyotacanada.scene7.com/is/content/toyotacanada/2025%20Corolla%20Hybrid%20Product%20Informationpdf?%24%3Fcdh=attachment&bfc=off&utm_source=chatgpt.com)
* **Electric motor power output (front) ~70 kW (93 hp).** [**toyotacanada.scene7.com**](https://toyotacanada.scene7.com/is/content/toyotacanada/2025%20Corolla%20Hybrid%20Product%20Informationpdf?%24%3Fcdh=attachment&bfc=off&utm_source=chatgpt.com)
* **Curb weights: e.g., 3,053 lb (~1,385 kg) for LE FWD version.** [**toyotacanada.scene7.com+1**](https://toyotacanada.scene7.com/is/content/toyotacanada/2025%20Corolla%20Hybrid%20Product%20Informationpdf?%24%3Fcdh=attachment&bfc=off&utm_source=chatgpt.com)
* **Fuel consumption (metric): e.g., 4.9 L/100 km combined for one version.** [**toyotacanada.scene7.com**](https://toyotacanada.scene7.com/is/content/toyotacanada/2025%20Corolla%20Hybrid%20Product%20Informationpdf?%24%3Fcdh=attachment&bfc=off&utm_source=chatgpt.com)

***1.12 (Gender Neutrality)***

Many people want to eliminate sexism in all forms of communication. You’ve been asked to create a program that can process a paragraph of text and replace gender-specific words with gender-neutral ones. Assuming that you’ve been given a list of gender-specific words and their gender-neutral replacements (e.g., replace both “wife” and “husband” with “spouse,” “man” and “woman” with “person,” “daughter” and “son” with “child”), explain the procedure you’d use to read through a paragraph of text and manually perform these replacements. How might your procedure generate a strange term like “woperchild?” You’ll soon learn that a more formal term for “procedure” is “algorithm,” and that an algorithm specifies the steps to be performed and the order in which to perform them. We’ll show how to develop algorithms then convert them to Java programs which can be run on computers.

***ANSWER***

** Start with a paragraph of text.  
Example:**

**“The man and his wife took their son to the park.”**

** Create a list (dictionary) of words to replace:**

| **Gender-specific word** | **Gender-neutral replacement** |
| --- | --- |
| **man** | **person** |
| **woman** | **person** |
| **husband** | **spouse** |
| **wife** | **spouse** |
| **son** | **child** |
| **daughter** | **child** |

** Read the paragraph word by word.**

** Check each word against your list of gender-specific words.**

** If a match is found, replace it with the gender-neutral word.**

** If no match, leave the word unchanged.**

** Continue until you reach the end of the paragraph.**

** Output the new, edited paragraph.**