**Chapter1:**

* Assignment answers:

a) Programs

b) Input unit, output unit, memory unit, arithmetic and logic unit (ALU), control unit, secondary storage

c) Machine language, assembly language, high-level language

d) Compilers

e) Android

f) Release candidate

g) Accelerometer

1.2:

Here are the answers:

a) java

b) javac

c) .java

d) .class

e) bytecode

1.3:

Here are the answers:

a) Information hiding

b) Classes

c) Object-oriented analysis and design

d) Inheritance

e) UML (Unified Modeling Language)

f) Attributes

1.4:

Here are the answers:

a) Input unit

b) Programming

c) Assembly language

d) Output unit

e) Memory unit and secondary storage

f) Arithmetic and logic unit (ALU)

g) ALU

h) High-level languages

i) Machine language

j) Control unit

1.5:

Here are the answers:

a) Java

b) C

c) Transmission Control Protocol (TCP)

d) C++

1.6:

Here are the answers:

a) Edit, compile, load, verify, execute

b) Integrated Development Environment (IDE)

c) Java Virtual Machine (JVM)

d) Virtual machine

e) Class loader

f) Bytecode verifier

**In Java, the compilation process occurs in two main phases:**

1. \*\*Source Code Compilation\*\*:

In the first phase, the Java compiler (\*\*javac\*\*) translates the human-readable Java source code (contained in files with the `.java` extension) into \*\*bytecode\*\*. Bytecode is an intermediate, platform-independent code stored in `.class` files. This step ensures that Java programs can run on any system with a compatible Java Virtual Machine (JVM).

2. \*\*Bytecode Execution\*\*:

In the second phase, the \*\*Java Virtual Machine (JVM)\*\* executes the bytecode. During this step, the JVM converts the bytecode into machine code that the specific computer's processor can understand. This involves a process called \*\*Just-In-Time (JIT) Compilation\*\*, which translates portions of bytecode into machine code at runtime to optimize performance.

Sure! Let's explore how these object-oriented programming (OOP) concepts can be applied to a wristwatch:

1. \*\*Object\*\*:

A wristwatch is an object, as it represents a tangible item with specific properties and functionalities.

2. \*\*Attributes\*\*:

These are the characteristics of the watch, such as its \*\*brand\*\*, \*\*material (e.g., leather strap, stainless steel case)\*\*, \*\*color\*\*, \*\*weight\*\*, \*\*battery life\*\*, and \*\*type (analog or digital)\*\*.

3. \*\*Behaviors\*\*:

The watch performs actions or has functionalities, such as \*\*displaying time\*\*, \*\*showing the date\*\*, \*\*setting alarms\*\*, or \*\*tracking steps (in smartwatches)\*\*.

4. \*\*Class\*\*:

A "Watch" class serves as a blueprint for creating various watches. It defines shared attributes (e.g., brand, type, material) and behaviors (e.g., show time, set alarm).

5. \*\*Inheritance\*\*:

Subclasses can inherit features from the "Watch" class. For example:

- An "AlarmClock" subclass inherits attributes (e.g., brand, material) and behaviors (e.g., displaying time) from the "Watch" class but adds its own unique behaviors like \*\*setting multiple alarms\*\*.

- A "Smartwatch" subclass inherits from the "Watch" class but includes advanced features like \*\*heart rate monitoring\*\* and \*\*Bluetooth connectivity\*\*.

6. \*\*Modeling\*\*:

Modeling involves creating a virtual representation of a watch to understand its design and functionality. This could include mapping out its \*\*attributes\*\* (e.g., size, color) and \*\*behaviors\*\* (e.g., how to set the time) before manufacturing it.

7. \*\*Messages\*\*:

Objects communicate by sending messages. For a smartwatch, a message could be the user tapping the screen to check their heart rate, which the watch processes and responds to by displaying the requested information.

8. \*\*Encapsulation\*\*:

Watches encapsulate their components (e.g., gears, battery, microchips) within a case to protect them. In programming terms, the internal workings (logic) of the watch are hidden, and only specific functionalities (e.g., time display) are exposed.

9. \*\*Interface\*\*:

The interface is how a user interacts with the watch, such as the \*\*dial\*\* (analog watches), \*\*buttons\*\*, or \*\*touch screen\*\* (digital and smartwatches). It provides access to certain features without revealing the internal workings.

10. \*\*Information Hiding\*\*:

Information hiding ensures that only essential details are exposed to the user while the complexity remains hidden. For example, a user can view the time on the watch face but doesn't need to understand how the internal mechanism (gears or circuits) works.

1.9:

Carbon Footprint = Activity Data × Emission Factor

An **emission factor** is a value used to estimate the amount of greenhouse gas (GHG) emissions produced per unit of a specific activity or input. It acts as a standardized metric to quantify emissions for various activities, making it easier to calculate carbon footprints.

1.10:

The formula for calculating Body Mass Index (BMI) is:

BMI = Weight (kg) / [Height (m)]²

Where:

* Weight is your mass in kilograms (kg).
* Height is your height in meters (m). The height is squared in the formula.

1.11:

Here are five popular hybrid vehicles and their key features:

1. Toyota Prius:

- Fuel Efficiency: 58 MPG city / 53 MPG highway.

- Powertrain: 1.8L 4-cylinder engine with an electric motor.

- Battery: Lithium-Ion or Nickel-Metal Hydride (depending on trim).

- Drive Modes: Eco, Power, and EV modes.

- Regenerative Braking: Recharges the battery during braking.

2. Honda CR-V Hybrid:

- Fuel Efficiency: 40 MPG city / 35 MPG highway.

- Powertrain: 2.0L 4-cylinder engine with two electric motors.

- Battery: Lithium-Ion.

- Drive Modes: Normal, Eco, and Sport.

- All-Wheel Drive\*\*: Standard on hybrid models.

3. Ford Maverick Hybrid:

- Fuel Efficiency: 42 MPG city / 33 MPG highway.

- Powertrain: 2.5L 4-cylinder engine with an electric motor.

- Battery: Lithium-Ion.

- Towing Capacity\*\*: Up to 2,000 lbs (4,000 lbs with optional package).

- Drive Modes\*\*: Normal, Eco, Sport, Slippery, and Tow/Haul.

4. Hyundai Tucson Hybrid:

- Fuel Efficiency: 38 MPG city / 38 MPG highway.

- Powertrain: 1.6L turbocharged engine with an electric motor.

- Battery: Lithium-Ion Polymer.

- All-Wheel Drive: Standard.

- Features: Panoramic sunroof, advanced safety systems.

5. Kia Sorento Hybrid:

- Fuel Efficiency: 39 MPG city / 35 MPG highway.

- Powertrain: 1.6L turbocharged engine with an electric motor.

- Battery: Lithium-Ion Polymer.

- Seating: Up to 6 passengers.

- Drive Modes: Eco, Sport, and Smart.

**Here's an algorithm for replacing gender-specific words with gender-neutral ones:**

**---**

### \*\*Algorithm\*\*

1. \*\*Input\*\*:

- Start with a paragraph of text.

- Create two lists or arrays:

- `genderWords` containing gender-specific words (e.g., "man", "woman", "wife", "husband", "daughter", "son").

- `neutralWords` containing the corresponding gender-neutral replacements (e.g., "person", "person", "spouse", "spouse", "child", "child").

2. \*\*Initialize Loop\*\*:

- Use a loop to iterate through all the words in the `genderWords` list.

3. \*\*Word Matching\*\*:

- For each word in `genderWords`:

- Check if the word exists in the paragraph.

4. \*\*Replacement\*\*:

- Replace every occurrence of the word in the paragraph with its corresponding word from the `neutralWords` list.

5. \*\*Output\*\*:

- After all replacements are complete, output the updated paragraph.

---

Simplified Example of the Algorithm

1. Input text: `"The man and woman are husband and wife."`

2. Lists:

- `genderWords = ["man", "woman", "husband", "wife"]`

- `neutralWords = ["person", "person", "spouse", "spouse"]`

3. Loop through `genderWords`.

4. For each word:

- Replace `"man"` with `"person"`.

- Replace `"woman"` with `"person"`.

- Replace `"husband"` with `"spouse"`.

- Replace `"wife"` with `"spouse"`.

5. Output: `"The person and person are spouse and spouse."`