TOUCHPAD

Information Technology (V.5.0)

7

Teacher's Manual

Extended Support for Teachers

ORANGE
www.orangeeducation.in

Teacher's Time Table

Sessions / Days	0	I	II	III	IV	BREAK	V	VI	VII	VIII
Monday										
Tuesday										
Wednesday										
Thursday										
Friday										
Saturday										

Development Milestones in a Child

Development milestones are a set of functional skills or age-specific tasks that most children can do at a certain age. These milestones help the teacher identify and understand how children differ in different age groups.

Age
5 - 8 Years

Physical	• First permanent tooth erupts • Shows mature throwing and catching patterns • Writing is now smaller and more readable • Drawings are now more detailed, organised and have a sense of depth
Cognitive	• Attention continues to improve, becomes more selective and adaptable • Recall, scripted memory, and auto-biographical memory improves • Counts on and counts down, engaging in simple addition and subtraction • Thoughts are now more logical
Language	• Vocabulary reaches about 10,000 words • Vocabulary increases rapidly throughout middle childhood
Emotional/ Social	• Ability to predict and interpret emotional reactions of others enhances • Relies more on language to express empathy • Self-conscious emotions of pride and guilt are governed by personal responsibility • Attends to facial and situational cues in interpreting another's feelings • Peer interaction is now more prosocial, and physical aggression declines

"If you cannot do great things, do small things in a great way."

Age 9 - 11 Years

Physical	• Motor skills develop resulting in enhanced reflexes
Cognitive	• Applies several memory strategies at once • Cognitive self-regulation is now improved
Language	• Ability to use complex grammatical constructions enhances • Conversational strategies are now more refined
Emotional/ Social	• Self-esteem tends to rise • Peer groups emerge

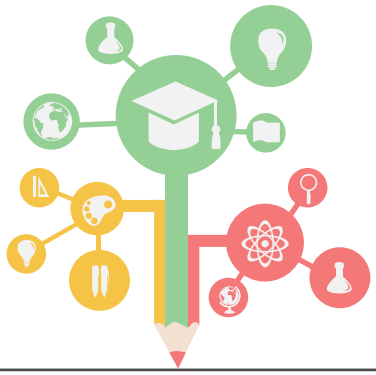
Age 11 - 20 Years

Physical	• If a girl, reaches peak of growth spurt • If a girl, motor performance gradually increases and then levels off • If a boy, reaches peak and then completes growth spurt • If a boy, motor performance increases dramatically
Cognitive	• Is now more self-conscious and self-focused • Becomes a better everyday planner and decision maker
Emotional/ Social	• May show increased gender stereotyping of attitudes and behaviour • May have a conventional moral orientation

Managing the children's learning needs according to their developmental milestones is the key to a successful teaching-learning transaction in the classroom.

"Family is the most important thing in the world."

Teaching Pedagogies



Pedagogy is often described as the approach to teaching. It is the study of teaching methods including the aims of education and the ways in which such goals can be achieved.

Lesson Plans

A lesson plan is the instructor's road map which specifies what students need to learn and how it can be done effectively during the class time. A lesson plan helps teachers in the classroom by providing a detailed outline to follow in each class.

A lesson plan addresses and integrates three key components:

- Learning objectives
- Learning activities
- Assessment to check the student's understanding

A lesson plan provides an outline of the teaching goals:

Before the class

1. Identify the learning objectives.
2. Plan the lesson in an engaging and meaningful manner.
3. Plan to assess student's understanding.
4. Plan for a lesson closure.

During the class

Present the lesson plan.

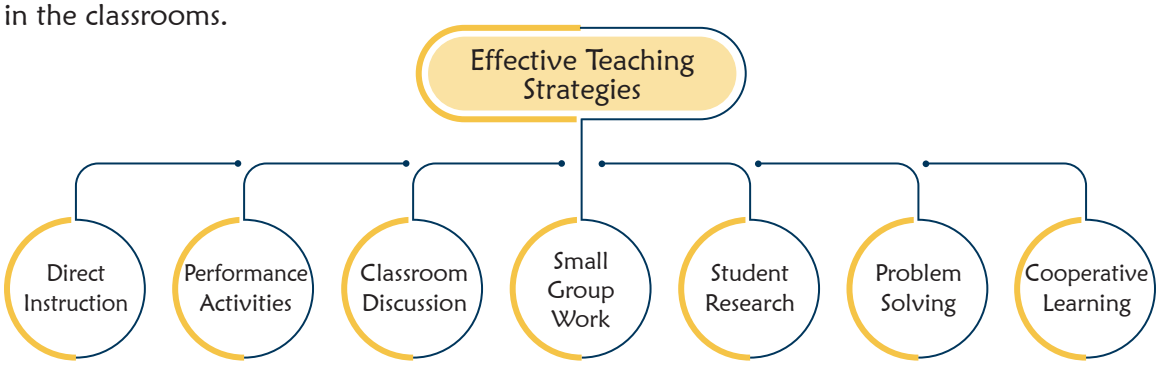
After the class

Reflect on what worked well and why. If needed, revise the lesson plan.

"Knowing yourself is the beginning of all wisdom."

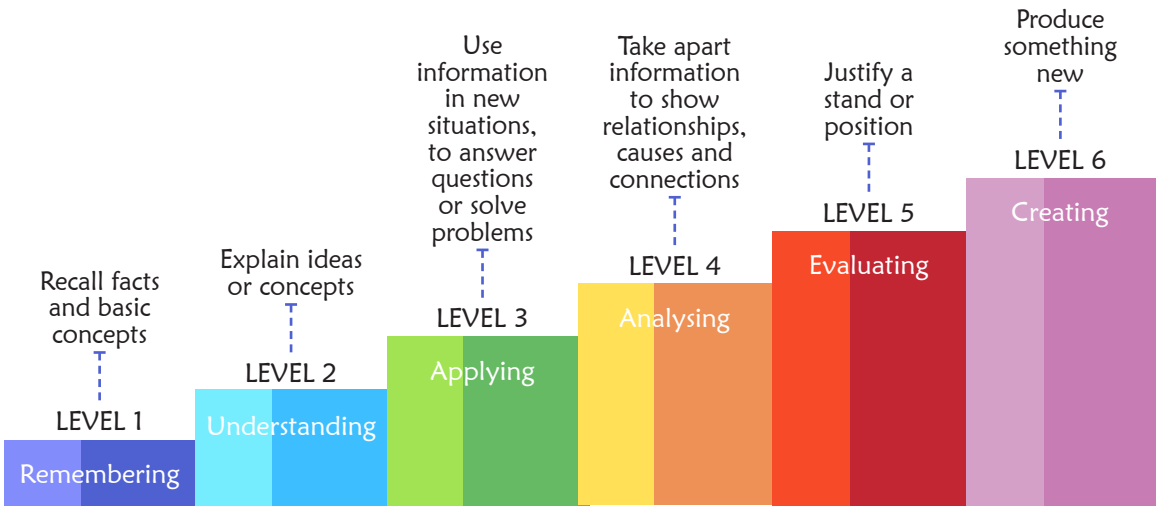
Teaching Strategies

Numerous strategies have evolved over the years to facilitate the teaching-learning process in the classrooms.



Bloom's Taxonomy

Bloom's Taxonomy was created by Dr Benjamin Bloom and several of his colleagues, to promote higher forms of thinking in education instead of rote learning. There are three domains of learning: cognitive (mental), affective (emotional), and psychomotor (physical). However, when we refer to Bloom's Taxonomy we speak of the cognitive domain. Bloom's Taxonomy is a list of cognitive skills that is used by teachers to determine the level of thinking their students have achieved. As a teacher, one should attempt to move students up the taxonomy as they progress in their knowledge.



Teachers should focus on helping students to remember information before expecting them to understand it, helping them understand it before expecting them to apply it to a new situation, and so on.

> "If you have no confidence in self, you are twice defeated in the race of life."

CLASS 7

LESSON PLAN

1 Fundamentals of Computer

Teaching Objectives

By the end of this lesson, students will be able to:

- Define what a computer is and explain its basic functions with emphasis on speed, accuracy, and versatility.
- Understand the components of a computer system (hardware and software) and their interactions through the IPO cycle (Input, Process, Output).
- Identify and explain various computer hardware components such as input, output, processing, and storage devices.
- Understand file management through File Explorer, and develop skills in creating, renaming, copying, and managing files and folders.
- Demonstrate how to use system utilities like Disk Cleanup and manage the Recycle Bin effectively.

Number of Sessions	
Theory	Practical
3	2

Teaching Plan

Engagement (Introduction)

1. **Question Prompt:** Ask students: "What types of activities do you perform on a computer every day?"
2. **Interactive Activity:** Introduce real-life scenarios (e.g., using a computer for homework, communication, and entertainment) and ask students to map these to the computer components involved (input, process, output).
3. **QAXP Activity:** In groups, students categorize various computer devices (input, output, storage devices) based on their knowledge and practical experiences.

Lesson Delivery

1. **What is a Computer?**
   - Define a computer and elaborate on its role in modern society as a versatile tool for work, entertainment, and communication.

- Discuss key characteristics of computers: **Speed**, **Accuracy**, **Storage**, **Versatility**, **Automation**, and **Diligence**.
- Think-Pair-Share: Ask students to give examples from their daily life where computers are helpful, and discuss how different components contribute to these tasks.

2. **Basic Computer Operations (IPO Cycle)**
   - Introduce the **Input-Process-Output (IPO) cycle** as the fundamental way computers handle data.
   - **Example:** Making orange juice – Input: Oranges, water, sugar; Process: Squeezing and mixing; Output: Orange juice.
   - Demonstration: Show how a computer uses the IPO cycle in tasks such as opening a document, performing calculations, and producing output on the screen.
   - Engage students in a Peer Teaching exercise where they explain the IPO cycle using other real-life examples.
3. **Computer Components**
   - **Hardware:** Discuss the different hardware components (Input, Processing, Output, Storage devices) using diagrams and actual devices (e.g., mouse, keyboard, CPU).
   - **Software:** Briefly explain the role of software in managing hardware and performing tasks.
   - **Round Robin Activity:** Groups identify and categorize hardware as input devices (keyboard, mouse), processing devices (CPU), output devices (monitor, printer), and storage devices (hard disk, SSD).
4. **File Management (File Explorer and Recycle Bin)**
   - **File Explorer:** Show students how to access and navigate File Explorer. Demonstrate how to search, create, rename, copy, and move files.
   - Discuss the importance of file naming conventions and the file extension system (e.g., .docx, .jpg).
   - **Recycle Bin:** Demonstrate how files are temporarily stored in the Recycle Bin and can be restored or permanently deleted.
   - **Practical Activity:** Guide students through tasks like creating a new folder, renaming a file, copying files, and restoring files from the Recycle Bin.

Practical Activities

- **File Management Practice:** Students will perform tasks such as creating folders, moving, renaming, and deleting files. They will also search for files using specific keywords and file extensions (e.g., *.docx).
- **Disk Cleanup & Recycle Bin Management:** Students will use Disk Cleanup to remove unnecessary files and organize data in the Recycle Bin.

- **Group Task:** Each group will use File Explorer to organise a sample folder structure and present their file management method.

Extension

- **Creative Task:** Students design a visual poster showcasing the various components of a computer system (input, output, processing, storage).
- **Discussion Prompt:** Ask students to reflect on how data is used and transformed by computers in their daily lives, e.g., gaming, school projects, and social media.
- **Oral Questions:**
  - What is the difference between hardware and software?
  - Explain the role of CPU in the IPO cycle.
  - Why is it important to back up files regularly?
  - How does Disk Cleanup help maintain a computer's performance?

Assessment and Evaluation

- **Quiz:** Multiple-choice and short-answer questions on the basic functions of a computer, components of the IPO cycle, and hardware/software distinction.
- **Oral Review:** Ask students to explain the IPO cycle and identify components of the computer system from a diagram.
- **Practical Test:** Students will search for a file, move it between folders, and restore it from the Recycle Bin. They will also perform Disk Cleanup.

Suggested Activity

- **Group Activity:** Organise a role-play where students act out the computer system components (e.g., CPU, memory, input devices, output devices).
- **Digital Drill:** In the computer lab, students will search for files using wildcard characters (e.g., *.jpg), organize their folders, and perform cleanup tasks.

2 Algorithm and Flowchar

Teaching Objectives

By the end of this lesson, students will be able to:

- Define algorithms and flowcharts, and explain their importance in problem-solving.
- Break down problems into smaller parts and create algorithms to solve them.
- Write algorithms using sequential, selection, and repetition control structures.
- Create flowcharts based on algorithms using correct symbols and logical flow.

- Differentiate between algorithms and flowcharts and understand when to use each.
- Use Google Drawings to create flowcharts based on given scenarios.

Number of Sessions	
Theory	Practical
3	2

Teaching Plan

Engagement (Introduction)

1. **Question Prompt:** "How do you approach solving problems in your everyday life? What steps do you follow?"
2. **Analogy Activity:** Compare creating an algorithm to writing a recipe for cooking.
3. **QAXP Strategy:** Students form groups and brainstorm a simple problem (e.g., preparing a sandwich) and how they would solve it step by step.

Lesson Delivery

1. **Introduction to Algorithms**
   - Define an algorithm: A step-by-step procedure or set of rules to solve a problem.
   - Discuss its importance in programming, especially for solving problems efficiently.
   - Example: Write an algorithm for making a sandwich.
   - Guidelines for Writing Algorithms: Begin with "Start", end with "Stop", written in simple, clear steps.
2. **Control Structures in Algorithms**
   - **Sequential:** Explain and demonstrate how instructions are executed in a sequence (e.g., calculating the area of a rectangle).
   - **Selection:** Introduce decision-making (e.g., checking if a number is divisible by 5).
   - **Repetition:** Discuss loops (e.g., displaying even numbers from 2 to 10).
   - **Peer Teaching:** In groups, students create examples using each control structure.
3. **Flowcharts**
   - Define flowcharts and their importance in visually representing algorithms.
   - Introduce basic flowchart symbols (Oval for start/stop, Parallelogram for input/output, Rectangle for process, Diamond for decision, Arrow for flow direction).
   - Example: Create a flowchart for calculating the area of a rectangle.
4. **Google Drawings for Flowchart Creation**
   - Guide students in using Google Drawings to create flowcharts based on given problems.
   - Demonstrate step-by-step how to use shapes and arrows to construct flowcharts.
   - **Practical Activity:** Students will create a flowchart for a task like crossing the road safely or calculating the area of a rectangle.

Practical Activities

- **Algorithm Writing:** Students write algorithms for everyday tasks (e.g., getting ready for school, calculating the average of three numbers).
- **Flowchart Creation:** Students use Google Drawings to create flowcharts based on algorithms they write (e.g., calculating the factorial of a number).
- **Group Project:** Each group is given a scenario (e.g., calculating the perimeter of a rectangle) and must write an algorithm and create a flowchart.

Extension

- **Creative Task:** Write an algorithm and create a flowchart for a real-world problem that the class can relate to (e.g., finding the best route to school).
- **Discussion Prompt:** Ask students to discuss which method (algorithm vs flowchart) would be better for solving a complex problem and why.
- **Oral Questions:**
    - What are the advantages of using an algorithm to solve a problem?
    - When should you use a flowchart over an algorithm?
    - How does repetition work in algorithms?

Assessment and Evaluation

- **Quiz:** Objective questions on control structures, flowchart symbols, and algorithm types.
- **Practical Test:** Write an algorithm for a given task and create a corresponding flowchart.
- **Oral Review:** Ask students to explain the steps in an algorithm and demonstrate the flow of a flowchart.
- **Peer Assessment:** Evaluate group projects on their algorithm-writing and flowchart-creation skills.

Suggested Activity

- **Flowchart Challenge:** In pairs, students are given a random task (e.g., preparing breakfast) and must quickly write an algorithm and flowchart.
- **Guessing Game:** Collaborate to design flowcharts for various real-world problems (e.g., managing school tasks or organizing a trip).

3 Learning Scratch

Teaching Objectives

By the end of this lesson, students will be able to:

- Understand the basics of Scratch programming and its visual interface.
- Create programs using blocks in Scratch to control sprite movement, appearance, and sound.
- Implement logic using control structures like loops and conditional blocks.
- Develop interactive programs by integrating user inputs like mouse clicks and keyboard keys.
- Create animations, games, and interactive stories using Scratch.

Number of Sessions	
Theory	Practical
3	3

Teaching Plan

Engagement (Introduction)

1. **Question Prompt:** "Have you ever played a computer game or watched an animation? How do you think they are created?"
2. **Demonstration:** Show a simple Scratch animation or game and explain how it works.
3. **QAXP Activity:** In groups, students brainstorm how they would create an interactive game using Scratch and share ideas with the class.

Lesson Delivery

1. **Introduction to Scratch**
   - Explain that Scratch is a visual programming language designed to help beginners create interactive programs.
   - **Discuss the Scratch interface: stage**, **sprites**, **blocks palette**, **coding area**.
   - **Example:** Create a simple project where the sprite moves across the screen.
   - **Think-Pair-Share:** Ask students to discuss how moving a sprite across the screen relates to programming.
2. **Understanding Scratch Blocks**
   - Introduce the different categories of Scratch blocks (e.g., motion, looks, sound, events, control, sensing, operators, variables).
   - **Example:** Show how to use motion blocks to move a sprite and control its direction.
   - **Activity:** Have students create a simple program to move a sprite forward when the green flag is clicked.

3. **Control Structures in Scratch**
    - **Loops:** Explain how repeat and forever blocks are used to repeat actions.
    - **Conditionals:** Introduce if and if-else blocks for decision-making.
    - **Example:** Use the repeat block to make the sprite move across the stage repeatedly.
    - **Think-Pair-Share:** Ask students to think of a situation where they might use a loop or condition in their own projects.
4. **Adding Sound and Visual Effects**
    - Demonstrate how to add sounds to a project using sound blocks.
    - Show how to change the sprite's size, visibility, or costume using looks blocks.
    - **Example:** Add a sound and change the sprite's appearance when it reaches a certain point on the screen.
5. **Interactive Programming with User Input**
    - Teach students how to use the sensing blocks to detect user interactions (e.g., mouse clicks, key presses).
    - **Example:** Make the sprite say "Hello" when the mouse pointer touches it.
    - **Practical Activity:** Students create a program where the sprite moves when an arrow key is pressed.

Practical Activities

- **Activity 1:** Create a simple animation where a sprite moves across the screen and changes costume or color when clicked.
- **Activity 2:** Program a sprite to play a sound when it touches a certain color or object.
- **Group Project:** In groups, students design and program an interactive story or a mini-game using Scratch.

Extension

- **Creative Task:** Have students add multiple sprites and backgrounds to their project, creating a short interactive animation or story.
- **Discussion Prompt:** Ask students how Scratch can be used to build more complex games or interactive applications.
- **Oral Questions:**
    - How can loops make your program more efficient?
    - What happens if you don't use the right sequence of blocks in Scratch?
    - How can you use user input to make your program interactive?

Assessment and Evaluation

- **Quiz:** Objective questions on the types of Scratch blocks and their functions.
- **Practical Test:** Students create a project where a sprite moves and interacts with user input, like mouse clicks or key presses.
- **Peer Assessment:** Students share their Scratch projects with peers and explain how they used different blocks to achieve their goals.

Suggested Activity

- **Game Creation Challenge:** Challenge students to design a simple game where the sprite has to collect objects or avoid obstacles on the stage.
- **Animation Showcase:** Have students create an animation using Scratch, where the sprite performs a series of actions triggered by the user.

4 Cyber Security

Teaching Objectives

By the end of this lesson, students will be able to:

- Understand the basic concept of computer networks and their types (LAN, WAN, PAN).
- Learn about the Internet and its importance in daily life.
- Identify common cybersecurity threats like malware, viruses, worms, and cybercrime.
- Understand and apply security practices such as using firewalls, antivirus software, and passwords.
- Recognize the significance of cyber laws and netiquettes in ensuring safe online behavior.

Number of Sessions	
Theory	Practical
3	2

Teaching Plan

Engagement (Introduction)

1. **Question Prompt:** "What would happen if our online accounts were hacked? How can we prevent this?"
2. **Interactive Activity:** Discuss the importance of passwords and encryption to protect online information.
3. **QAXP:** In groups, students list common online activities (e.g., shopping, social networking) and identify potential security risks.

Lesson Delivery

1. **What is a Network?**
   - Define a computer network: A group of connected computers and devices that allow people to communicate and share resources.

- Explain different types of networks: LAN (Local Area Network), WAN (Wide Area Network), PAN (Personal Area Network).
- **Example:** Use a simple classroom network model to show how devices communicate within a school or home.

2. **What is the Internet?**
   - Define the Internet: A global network that connects millions of computers and devices worldwide.
   - Discuss its applications in communication (email, social media), entertainment (streaming, gaming), e-commerce, and education.
   - **Activity:** Ask students to list some websites they visit frequently and discuss how they use the Internet in daily life.
3. **Cybersecurity Basics**
   - **Define cybersecurity:** The practice of protecting computers, networks, and data from unauthorized access, attacks, or damage.
   - Introduce common security threats: **Malware**, **Viruses**, **Worms**, **Trojans**, **Phishing**, **Spam**.
   - **Example:** Explain how a computer virus spreads and disrupts normal operations (e.g., slow performance, file corruption).
4. **Protecting Your Data: Tools and Practices**
   - **Antivirus Software:** Explain how antivirus programs protect against malicious software. Demonstrate installing and running antivirus scans.
   - **Firewalls:** Discuss the role of firewalls in blocking unauthorized access to systems. Show how to configure basic firewall settings.
   - **Passwords and Encryption:** Explain the importance of strong passwords and encrypting sensitive data. Discuss techniques for creating secure passwords.
5. **Cybercrime and Legal Protection**
   - **Define cybercrime:** Criminal activities that involve computers or networks (e.g., hacking, identity theft, cyberstalking).
   - **Case Study:** Discuss real-world examples of cybercrime, like phishing attacks or ransomware.
   - **Cyber Laws:** Explain the importance of cyber laws in protecting users and organizations from digital crimes. Discuss the Information Technology Act of India.

Practical Activities

- **Activity 1:** Students will simulate a simple network by connecting multiple devices (computers, smartphones) in a small network and practice sharing files and information.
- **Activity 2:** Using antivirus software, students will run a basic system scan on their computers or devices to identify and remove any threats.

- **Group Project:** Students will work in groups to create a security protocol for their online accounts, including password creation, two-factor authentication, and regular updates.

Extension

- **Creative Task:** Have students create a **Cybersecurity Poster** that explains common internet safety tips, such as avoiding suspicious links and using strong passwords.
- **Discussion Prompt:** How do cybercrimes affect both individuals and organizations? How can you protect yourself against these threats?
- **Oral Questions:**
  - What is the role of firewalls in protecting data?
  - How do phishing scams trick users into sharing personal information?
  - What is the difference between ethical and unethical hacking?

Assessment and Evaluation

- **Quiz:** Multiple-choice and short-answer questions on types of networks, internet safety tools, and common cybersecurity threats.
- **Practical Test** Students demonstrate securing an account by changing passwords, enabling two-factor authentication, and running an antivirus scan.
- **Peer Review:** Students present their security protocols for feedback from peers.

Suggested Activity

- **Simulation Exercise:** Students will act as different components of a network (e.g., devices, firewalls, servers) to simulate a network's operation and security protocols.
- **Interactive Task:** In the computer lab, students will identify and report potential cybersecurity threats in their current online practices (e.g., unsafe websites, weak passwords).