Teaching English to Computer Science Students: A Comprehensive Approach

IMPRESSUM

Title: *Teaching English to Computer Science Students: A Comprehensive Approach*

Authors: Dragana Božić Lenard

Graphic design & layout: Dino Žilić

Publisher: Cambridge Scholars Publishing

Year of publication: 2025

ISBN:

Contributors
Subject matter experts:
Dr. Ivan Aleksi, Associate Professor, Josip Juraj Strossmayer University of Osijek, Croatia
Dr. Zdravko Krpić, Associate Professor, Josip Juraj Strossmayer University of Osijek, Croatia
Dr. Krešimir Romić, Assistant Professor, Josip Juraj Strossmayer University of Osijek, Croatia
Josip Zidar, Master of Computer Engineering, Josip Juraj Strossmayer University of Osijek, Croatia

Peer reviewers:
Dr. Vesna Bulatović, Assistant Professor, University of Novi Sad, Serbia
Dr. Gabriela Chmelikova, Associate Professor, Slovak University of Technology in Bratislava, Slovakia
Dr. Violeta Jurković, Associate Professor, University of Ljubljana, Slovenia

Printing: Cambridge Scholars Publishing

Edition: First edition

Acknowledgments:
Special thanks to the Faculty of Electrical Engineering, Computer Science and Information technology Osijek, Croatia and students for their valuable feedback and contributions to the development of this coursebook.

Contact information:

Faculty of Electrical Engineering, Computer Science and InformationTechnology Osijek, Kneza Trpimira 2b, 31 000 Osijek

dragana.bozic@ferit.hr

https://www.ferit.unios.hr/2021/djelatnik.php?aai=d1bozic&ime=dragana-bozic-lenard&crosbi=&matbroj=249&domaci=1

Teaching English to Computer Science Students: A Comprehensive Approach

By

Dragana Božić Lenard

Cambridge Scholars Publishing

Teaching English to Computer Science Students: A Comprehensive Approach

By Dragana Božić Lenard

This book first published 2025

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN: 978-1-0364-5458-6
ISBN (Ebook): 978-1-0364-5459-3

TABLE OF CONTENTS

BOOK MAP

UNIT	READING	LISTENING/ WATCHING	WRITING	SPEAKING	LANGUAGE FOCUS	VOCABULARY
History of computing **5-35**	Matching headings **Focus: Reading for gist/skimming** Sequencing **Focus: Reading for details** Gap filling **Focus: Reading for contextual information**	Matching **Focus: Listening for gist** True/false **Focus: Listening for details** Gap filling **Focus: Listening for specific information**	*Who am I? game* Research presentation Role-play *Cluedo*	Jigsaw discussion	Past Simple vs Past Continuous	Word jumble race Bingo (antonyms)
Computer users **36-59**	Word formation Synonyms and antonyms **Focus: Reading for vocabulary development**	Table completion, matching people with habits **Focus: Listening for specific information** Matching causes and effects, Venn diagram **Focus: Listening for details**	Reflection from a given viewpoint Public awareness campaign poster, slogan or infographic	Role-play Lightning talk Stakeholder-based debate	Past Simple vs Past Continuous	Word clusters Keyword-based paraphrasing
Computer architecture **60-95**	Summarizing and paraphrasing **Focus: Reading for key information and details** Quiz drafting **Focus: Reading for understanding**	Table completion **Focus: Listening for details** Role-play **Focus: Listening for contextual understanding** Error-correction **Focus: Listening for accuracy**	Summarizing infographic Comparison and contrast advisory report	Comparison and contrast expressions Storytelling	Past Simple vs Present Perfect Simple Role-play	Taboo (vocabulary item definition) Word maze vocabulary item recognition) Yes/no quiz on computer parts Jigsaw puzzle (vocabulary item contextualization)

UNIT	READING	LISTENING/ WATCHING	WRITING	SPEAKING	LANGUAGE FOCUS	VOCABULARY
Storage **96-119**	Matching headings **Focus: Reading for gist/skimming** Error correction **Focus: Reading for details** Analogies **Focus: reading for identifying relationships between concepts**	Matching attributes **Focus: Listening for gist** Gap filling **Focus: Listening for specific information** Sequencing **Focus: Reading for details**	Recognition of and rewriting topic sentences Linking words	Case study	Phrasal verbs	True/false Word formation One word fit three gaps Odd one out
Operating systems **120-145**	Jigsaw reading **Focus: Reading for collaborative learning** Word clustering and matching **Focus: Reading for vocabulary development**	Multiple-choice and yes/no question quiz **Focus: Listening for specific information** Phrasal verbs **Focus: Listening for vocabulary development**	Paraphrasing domino game Paraphrasing techniques	Think-pair-share Role-play Case study discussion	Past and present tenses	Phrasal verbs One word fit three gaps Odd one out *Who wants to be a millionaire* quiz
Internet **146-170**	Short answer questions Paragraph topics True/false **Focus: Reading for gist (skimming and scanning)**	Reflection essay **Focus: Listening for gist (global understanding)**	Reflection essay Fact-checking Creative writing	Speed-chat Debate	Expressing future (Present Simple, Present Continuous, Future Simple or Going to Future)	Memory game (antonyms, definitions) Adjectives vs adverbs Word building (noun/verb/ adjective/ adverb) Odd one out Word transformation

USER GUIDE

USER GUIDE

Welcome to the *Teacher's Resource Book* for *English for Computer Science – part I*, a carefully curated companion to the student coursebook aimed at university-level learners in computer science/ engineering, information technology, and related fields. This resource is developed with practicing teachers in mind, providing a structured, practical, and pedagogically sound framework for delivering effective and engaging lessons in English for Specific Purposes (ESP), particularly in the field of computer science.

As digital technologies continue to reshape our world, the ability to communicate fluently and accurately about technical topics in English has become essential. This resource book addresses that need by providing ESP teachers with tools and strategies to make technical English instruction both accessible and stimulating for diverse learners.

The book follows the structure of the student coursebook, comprising six content-rich units:
1) History of computing
2) Computer users
3) Computer architecture
4) Storage
5) Operating systems
6) Internet

Each unit builds on the content presented in the coursebook and is enhanced with a variety of resources to facilitate teaching, support student learning, and foster the development of both language proficiency and subject knowledge.

Purpose and pedagogical rationale
The content is designed to facilitate not only language acquisition but also the development of 21st-century skills such as collaboration, problem-solving, critical thinking, and project-based learning. Whether you are an experienced ESP teacher or new to teaching technical content in English, this book equips you with adaptable resources for planning and delivering lessons with confidence.

The *Teacher's Resource Book* has been created with the following goals:
- To **provide additional practice opportunities** that consolidate and extend learners' language and content knowledge;
- To **equip teachers with strategies** for handling common classroom challenges, especially in mixed-ability and multilingual environments;
- To **promote active learning** through project-based tasks and interactive methodologies;
- To support **formative assessment** through quizzes and continuous evaluation tools;
- To offer **ready-to-use lesson plans and teaching tips** that reduce preparation time while ensuring pedagogical quality.

Structure and components of each unit
Each unit in the *Teacher's Resource Book* contains the following components:
1) Additional exercises
To deepen students' understanding and consolidate key vocabulary, structures, and content, each unit includes supplementary tasks that build on the material in the student's book. These exercises may include gap fills, matching tasks, speaking prompts, short writing tasks, mini case studies, and reading extensions, designed to enhance both language and technical knowledge. They come with their respective key.

These activities complement the student book by providing further opportunities to practice:
- **Key vocabulary** in context, with a focus on collocations, affixation, and word families;
- **Grammar structures** relevant to the unit's topic (e.g. past tenses for historical overviews);
- Reading and listening **comprehension tasks** with authentic or semi-authentic texts;
- **Functional language** for academic and professional settings, such as explaining processes, comparing systems, or making predictions.

2) Teacher's notes and lesson plan
This section is designed to provide practical guidance, pedagogical support, and ready-to-use lesson outlines to help teachers deliver content confidently and effectively. It serves both novice and experienced teachers by offering structure while allowing flexibility for individual teaching styles and classroom contexts.

Each unit includes a **detailed lesson plan and accompanying notes**, featuring the following components:
- **Lesson objectives** and **outcomes** aligned with both language and content outcomes;
- **Estimated timing** and pacing suggestions;
- **Materials and resources** needed (e.g. multimedia, student worksheets);
- **Suggested procedures** for warm-up, input, guided practice, production, and wrap-up;
- **Notes on classroom management**, transitions, and time-saving strategies.

3) Teaching strategies and classroom management
This section supports teachers with a set of adaptable, practical techniques that enhance engagement, promote effective learning, and maintain a positive classroom environment in English for Computer Science courses. These strategies are especially useful for handling challenges specific to mixed-ability groups, varying levels of digital and linguistic literacy, and diverse learning styles. Special attention is given to fostering a communicative and inclusive classroom atmosphere.

This section provides overarching guidance for effective teaching, including:
- Strategies for engaging students through task-based language teaching, flipped classroom, project-based and active learning;
- Managing pair and group work in a technical ESP context;
- Encouraging autonomous learning and digital tool integration;
- Handling fast finishers and disengaged learners;
- Managing classroom language and code-switching effectively.

4) Common challenges and solutions
This section identifies frequent learner difficulties related to the unit content, whether linguistic (e.g. understanding specialized terminology), conceptual (e.g. abstract technical processes) or skill-based (e.g. giving presentations or explaining a process). Practical, classroom-tested solutions are provided to help teachers pre-empt and address these challenges effectively.

This section anticipates linguistic, cognitive, and affective challenges your learners may encounter, such as:
- Confusion between similar technical terms;
- Difficulty expressing abstract or logical relationships in English;
- Uneven language proficiency levels;
- Lack of motivation when faced with dense or abstract content.

For each challenge, the book offers practical solutions, including:
- Visual aids, tiered and scaffolding techniques;
- Guided discovery activities;
- Integrating multimedia resources;
- Peer teaching and collaborative tasks;
- Use of first-language support where appropriate;
- Incorporating everyday analogies and creating code-switching activities.

5) Differentiation tips for mixed-ability classes
Recognizing the diversity of learners in today's classrooms, this section offers strategies for differentiating content, process, and product to meet the needs of students at varying proficiency levels.

Given the diversity of language levels and learning styles in most classrooms, this section offers tailored strategies such as:
• Tiered tasks and scaffolded support with varying levels of complexity;
• Optional challenge questions for advanced learners;
• Additional support materials (e.g. glossaries, sentence frames) for less confident students;
• Strategies for flexible grouping and peer support.
The aim is to promote **inclusive learning** and ensure that all students are both supported and challenged.

6) Mini project ideas
Each unit includes suggestions for short, hands-on projects that promote collaborative learning and allow students to apply what they have learned in real-world contexts. These mini projects can be used as class-based activities, homework assignments, or formative assessments. Projects vary in format - presentations, posters, infographics, digital stories, simple coding tasks, mock interviews, user guides, etc. - and include suggested outcomes, materials, and options for differentiation. They aim to boost motivation, autonomy, and creativity as well as encourage critical thinking and real-world application of language and content.

Mini projects are ideal for:
• Reviewing and applying unit content in a meaningful context;
• Encouraging student collaboration and autonomy;
• Supporting project-based and task-based learning approaches;
• Offering opportunities for cross-curricular integration (e.g. combining ICT and English);
• Catering to different student interests, learning styles, and ability levels.

7) Unit quiz and answer key
To help assess students' progress and consolidate learning, each unit concludes with a short quiz testing both language and content knowledge. These quizzes are ready to use for quick assessments or revisions, and are accompanied by clearly presented answer keys.

Each quiz is balanced in terms of **language and content knowledge**, and may include the following task types:
• **Vocabulary and terminology** (e.g. matching, gap filling, paraphrasing or using word in contexts)
• **Grammar in context** (e.g. multiple-choice, sentence transformation tasks or error correction)
• **Reading or listening comprehension** (e.g. true/false, short-answer comprehension questions)
• **Writing tasks** (e.g. brief paragraph writing).

The *Key* following each quiz provides:
• **Correct answers** to objective questions (e.g. multiple choice, matching, gap-fills, etc.);
• **Model answers or suggested responses** for open-ended tasks (reading comprehension and writing);
• **Explanations or teaching tips** where needed, especially for tricky grammar or commonly confused terms;
Alternative acceptable answers for open-ended tasks to support classroom flexibility.

Flexibility and use
• Quizzes are printable and ready-to-use in class or as take-home tests
• Teachers can administer them individually, in pairs, or as interactive games/quizzes using digital tools
• Ideal for checking comprehension before moving on, as revision, or for formative assessment reports

How to use this book

This *Teacher's Resource Book* is designed to be flexible. It can be used in the following ways:

- **During lesson preparation**: To plan lessons more efficiently using the suggested activities, differentiation strategies, and teaching tips.
- **In the classroom**: To support the delivery of content-rich and language-focused instruction with ready-made exercises and mini projects.
- **For ongoing professional development**: To reflect on challenges in ESP teaching and explore alternative strategies and classroom management techniques.
- **For assessment**: To quickly check understanding through unit quizzes and integrate formative assessment into your teaching.

HISTORY OF COMPUTING

Task 1. Read the text on the milestones in computing history and match the titles (a-j) with their respective paragraphs (1-5). There are more titles than you need.

a) Bringing computers to the masses
b) The rise of user-friendly operating systems
c) Connecting the world through the web
d) Microprocessors and the birth of the PC
e) The cloud computing transformation
f) The open-source movement
g) Apple brings GUI to the masses
h) The smartphone revolution
i) Data centers and global connectivity
j) The era of personal computing

1) _____
In the early 1970s, the invention of the microprocessor marked a breakthrough in computing technology. The Intel 4004, created in 1971, was the first commercially available microprocessor, capable of processing information on a single chip. This development enabled smaller and more affordable computers, laying the foundation for the personal computer revolution.

2) _____
In the 1990s, the World Wide Web transformed computing once again by making the internet accessible to the public. Invented by Tim Berners-Lee in 1989, the web used a system of hyperlinks and web pages, allowing information to be shared globally. By 1995, the web had become a central feature of modern computing and everyday life.

3) _____
In 1991, Finnish student Linus Torvalds created Linux, a free, open-source operating system. Linux revolutionized software development by allowing anyone to access, modify, and distribute the source code. It became particularly popular for servers and continues to be widely used in various applications, from smartphones to supercomputers.

4) _____
In 2007, Apple's iPhone changed the mobile phone landscape, introducing the first smartphone with a multi-touch screen and user-friendly design. The iPhone combined phone, internet, and multimedia capabilities in one device, paving the way for the app economy and transforming how people communicate, work, and entertain themselves on the go.

5) _____
As cloud technology advanced in the 2010s, cloud computing became a major part of the tech industry. Platforms like Amazon Web Services (AWS) and Microsoft Azure allowed individuals and businesses to store and process data online instead of on local machines. Cloud computing enabled more flexibility, collaboration, and scalability in data storage and processing.

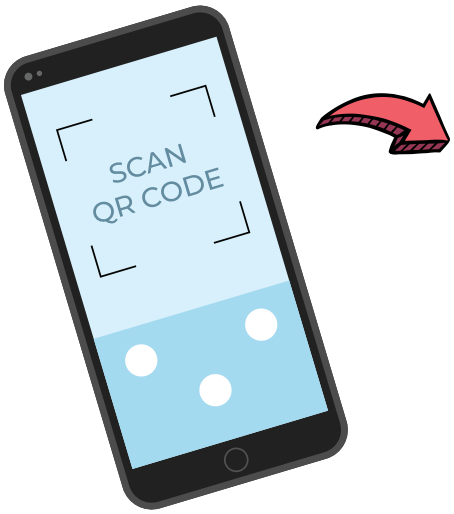
Task 2. Read the text again and order the events chronologically. Focus only on the events mentioned in the text. There are more events than you need.

d) The creation of the first microprocessor _____
g) Development of the Xerox Alto, the first GUI computer _____
c) Apple releases the first Macintosh _____
e) The launch of the world wide web _____
a) Creation of Linux _____
b) The release of Windows 95 _____
h) Invention of cloud computing technology _____
f) Introduction of the iPhone _____

Task 3. Complete the summary on the history of smartphone development with one word only.

The development of smartphones began in the 1) _____ 2000s, but it was Apple's release of the iPhone in 2007 that truly revolutionized the market. The iPhone combined a touch-screen interface with the ability to 2) _____ the internet, play multimedia, and run various applications. This innovation marked the shift 3) _____ basic mobile phones to smart devices that could serve as personal organizers, cameras, and gaming consoles. Following Apple's success, Google launched 4) _____ Android operating system 5) _____ 2008, providing an open-source alternative that quickly gained popularity. Android 6) _____ many companies to enter the smartphone market, leading to increased competition and a 7) _____ of options for consumers.

Task 4. Before listening, pair up, scan the QR code and match programming languages with their descriptions. After listening, check your matching performance.



Task 5. Listen and decide whether the statements are true (T) or false (F). If false, correct them.

1) COBOL, developed in 1959, is still in use today for business applications. T / F

2) C was the first high-level programming language. T / F

3) JavaScript and Java were both released in 1995 and became essential for web development. T / F

4) Swift was developed by Google to simplify Android development. T / F

5) Go, developed in 2009, is primarily used for web development. T / F

Task 6. Listen again and complete the gaps with one word only.

1) In 1957, _____ was developed by IBM and became one of the first high-level programming languages for _____ and engineering applications.

2) The 1980s saw the rise of object-oriented programming with _____, which helped organize code into reusable _____.

3) _____, developed in the 1970s, became the standard language for managing and retrieving data from _____.

4) The language _____, released in 1991, is known for its readability and support for multiple _____ styles.

5) Microsoft launched _____ in 2000 within the .NET framework, targeting _____ applications.

6) Recently, _____ has gained popularity for its emphasis on _____ safety and efficiency, making it suitable for system-level programming and game development.

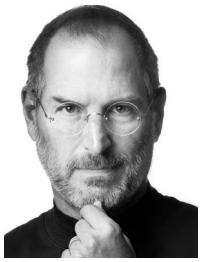
Task 7. Participate in the *Who am I?* game. Research your character's contributions and prepare a one-minute introduction into the character. Answer yes/no questions until your colleagues guess who you are.

Ada Lovelace

Alan Turning

Grace Hopper

John von Neumann



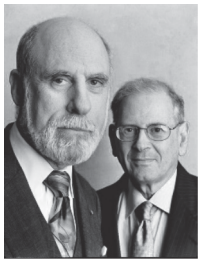
Steve Jobs

Bill Gates

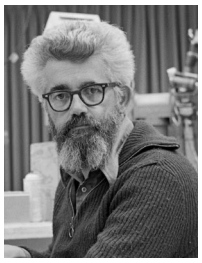
Tim Berners-Lee

Linus Torvalds

Mark Zuckerberg



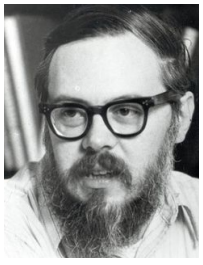
Vint Cerf and Bob Kahn



John McCarthy

Seymour Cray

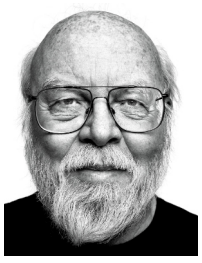
Niklaus Wirth



Dennis Ritchie

Guido van Rossum

Steve Wozniak



James Gosling

Evelyn Boyd Granville

Task 8. Study a decade/era in computing. Research inventions, key figures and major developments. Present your findings to the class and add it to the timeline drawn on the board.

1940s	1950s	1960s
1970s	1980s	1990s
2000s	2010s	2020s

Task 9. Pair up. One student is a time traveler from a distant past and another a modern day expert. Role play.

You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify the internet.	You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify smartphones.
You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify artificial intelligence.	You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify cloud computing.
You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify online shopping and e-commerce.	You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify social media.

You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify smart homes.	You are a time traveler from a distant past (e.g. the 1800s or early 1900s) who has arrived in the present day and is astounded by modern technology. You want to understand how people use technology and what it means for society. Ask questions.	You are a modern day "expert" who needs to use simplified language to explain unknown terminology to a time traveler from a distant past. Your task is to explain and exemplify 3D printing.

Task 10. Participate in a detective game *Cluedo*.

A renowned computer expert, Dr. Victor Carver, has been found dead in his high-tech office. He was sitting at his desk with a code fragment open on his screen and several suspicious files open on his computer. The detectives need to gather clues, talk to potential witnesses, and solve the mystery behind his death. Was it a rival trying to steal his work? Or was it someone closer to him with a hidden motive?

Scene: Dr. Carver's Office

The room is dimly lit with glowing computer screens displaying lines of code. The faint hum of machines can be heard in the background. Dr. Carver's chair is pushed back slightly, but his lifeless body is slumped over the desk, staring at his computer.

On the screen, a fragment of code appears to be unfinished. Nearby, a folder with the title "Confidential Research" is open, filled with encrypted files. The room is sleek and modern, but there's a tension in the air as everyone looks around suspiciously.

Detective Riley	**Sophie Mitchell**	**Jack Lawson**
Role: The lead detective investigating the case. Your job is to question everyone, analyze the clues, and piece together the puzzle. Objective: Find out who the murderer is by asking the right questions and looking for contradictions.	Role: Dr. Carver's research assistant. She worked closely with him on new programming languages. She might know more about his work and personal life. Objective: Tell what you know, but keep an eye on your emotions, as your relationship with Dr. Carver was close.	Role: A former colleague and rival in the field of computer programming. He and Dr. Carver often argued over who created the best programming languages. Objective: You have your own ideas about Dr. Carver's work. Show how your rivalry may have led to conflict, but do not give away too much.

Avery Park

Role: A cybersecurity expert who frequently collaborated with Dr. Carver. Avery knew about the files Dr. Carver was working on.

Objective: Keep your cool and try to help the detectives by sharing your knowledge of Dr. Carver's recent projects. But, you might have a secret of your own.

Maya Delgado

Role: Dr. Carver's partner. They were working together on a breakthrough project that could change the tech world. Maya had a lot to gain from the success, but also might have secrets.

Objective: You are devastated by the loss. Your role is to share personal details and try to cover any possible motives that could make you look guilty.

Lucas Green

Role: The building's IT technician. Lucas had access to Dr. Carver's office for maintenance and troubleshooting, especially for security systems.

Objective: Lucas can reveal details about who accessed the office and when. He might have noticed unusual activity but could be hiding something if he was involved.

Lena Price

Role: Dr. Carver's personal assistant. She managed his schedule, including secret meetings with corporate representatives and government officials interested in his work.

Objective: Lena knows about Dr. Carver's projects and can offer insight into his private meetings and confidential matters. She may have kept secrets that others do not know.

Damien Ward

Role: A hacker known to have had contact with Dr. Carver. Damien often consulted with Dr. Carver about security issues and ethical hacking techniques.

Objective: Damien might know about vulnerabilities in Dr. Carver's systems. He may have been interested in the files on Dr. Carver's computer, potentially for illegal use.

Oliver Banks

Role: The CEO of a major tech firm with a vested interest in Dr. Carver's research. Oliver had been funding Dr. Carver's latest project, hoping for a profitable outcome.

Objective: As someone who might lose money due to Dr. Carver's death, Oliver could have a motive, especially if there was a risk of the research not delivering as expected.

Rachel Yang

Role: Dr. Carver's former protégé, who recently split from his team to start her own tech company. There were rumors she left after a disagreement over intellectual property.

Objective: Rachel has a lot of technical knowledge and insight into Dr. Carver's work. However, her departure from his team was tense, and she may feel resentful or wronged.

Olivia Grant

Role: Olivia is an HR manager responsible for handling internal disputes, disciplinary actions, and performance reviews.

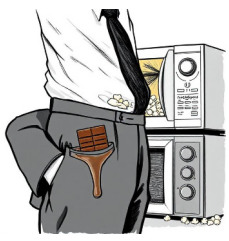
Objective: Olivia and Dr. Carver were close friends (and more than that) in university. Over the last year, Dr. Carver began distancing himself from Olivia, allegedly preparing to report her for unethical hiring practices. She, in turn, began pushing back on his lab's staffing decisions, blocking his requests for certain hires and initiating an internal investigation into lab conflicts.

ROLES

Encrypted email An email from an anonymous sender was found in Dr. Carver's inbox. The subject reads, "Urgent: They're on to us." The contents of the email are encrypted, hinting at a hidden project or collaboration Dr. Carver was involved in	**Missing server password** The password to one of Dr. Carver's private servers is missing from his usual notes. It's known that only Dr. Carver, his assistant Lena, and his partner Maya had access to it. This server may hold the key to understanding his latest project.	**Mysterious USB drive** A small USB drive was found on the floor near Dr. Carver's chair. When plugged into a computer, it shows a series of deleted files with titles like "Project Phoenix" and "Backdoor Prototype."
Handwritten note A crumpled note in Dr. Carver's handwriting reads, "Trust no one." Next to it, a list of initials includes those of some of his colleagues, but not all. This may imply distrust or even betrayal among his closest connections.	**Access log** A record from the building's security system shows someone accessed Dr. Carver's office late at night. The log indicates the use of a special access code known only to a few close associates.	**Unusual phone records** Dr. Carver made multiple calls to a number registered to an encrypted messaging service a few days before his death. Records show he communicated with this contact frequently, though no one knows who it was.
Piece of torn fabric A small, torn piece of fabric was found on the edge of Dr. Carver's desk. The color does not match any clothing found in the office, suggesting it might belong to the intruder.	**Financial records** Recent bank transactions show a large, unexplained withdrawal from Dr. Carver's account. The transaction appears suspicious, especially since he rarely made such large withdrawals.	**Code analysis** The code fragment open on Dr. Carver's screen contains a security flaw that could be exploited by hackers. Detectives might wonder if this was intentional or if someone was trying to sabotage his work.
Fingerprint smudges on screen Faint fingerprints were found on the screen where the code fragment was open. They do not match Dr. Carver's prints, suggesting someone else touched the screen.	**Coffee cup with lipstick smear and traces of sedative** A half-finished coffee cup was found on Dr. Carver's desk. Forensics identified traces of a mild sedative mixed in with the drink.	CLUES

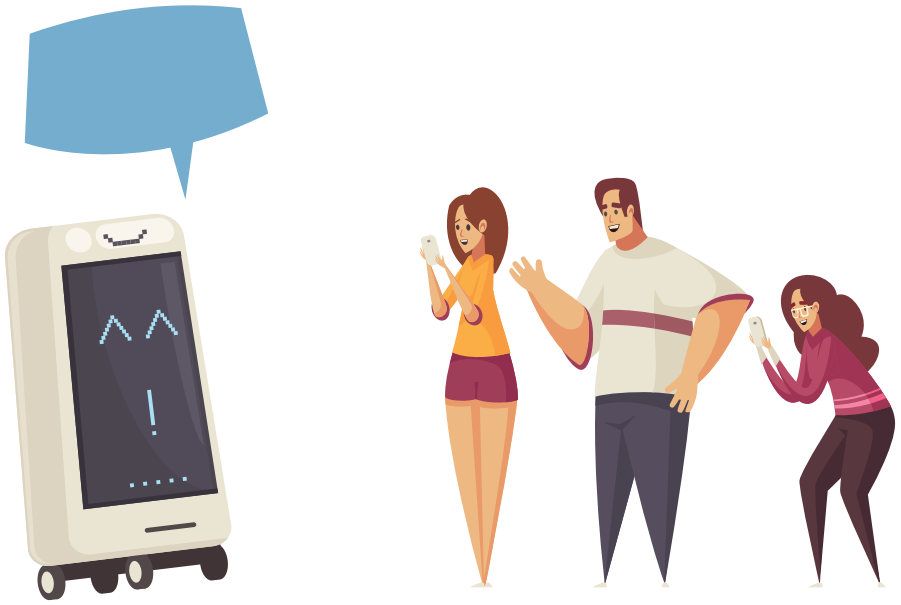
Task 11. Study the pictures provided by your instructor and describe what you think happened in the scene using past tenses.

a) 

b) 

c) 

Task 12. Compose a sentence describing something that (did not) happen to you at some point in the past. The class will have to decide whether it is true or call your bluff.



Task 13. Arrange the key milestones in the history of computing and technology in chronological order to create a timeline. Write the correct sequence in your notebook or on a timeline chart. Using the arranged timeline, write a short paragraph for each event, using past simple for completed actions and past continuous to describe any background activities or ongoing developments at the time.

EVENT CARDS (set A)	**1945 – ENIAC was completed** The first electronic general-purpose computer, ENIAC, was completed by scientists at the University of Pennsylvania. It could perform thousands of calculations per second, marking a significant leap in computing power.	**1969 – ARPANET was launched** The Advanced Research Projects Agency Network (ARPANET) was established, becoming the first network to connect multiple computers. This project laid the .
1971 – The first microprocessor was introduced Intel released the Intel 4004, the world's first microprocessor, allowing computers to become smaller and more powerful over time.	**1989 – The World Wide Web was proposed** Tim Berners-Lee introduced the concept of the World Wide Web, which would allow information to be shared across the internet using hyperlinks.	**2007 – The iPhone was released** Apple introduced the iPhone, a revolutionary smartphone with a touchscreen interface that combined internet access, multimedia, and communication.

EVENT CARDS (set B)

1956 – The first hard disk drive was created

IBM developed the first hard disk drive, the IBM 305 RAMAC, which could store 5 megabytes of data. Although it required an entire room for storage, this invention marked the beginning of modern data storage.

1973 – The Xerox Alto was introduced

Xerox developed the Alto, the first computer with a graphical user interface (GUI). Though it was not commercially released, it introduced features like windows, icons, and menus that would later influence personal computers.

1991 – Linux was created by Linus Torvalds

Linus Torvalds released the first version of Linux, an open-source operating system. This led to widespread collaboration in software development and made Linux a foundation for many systems today.

2004 – Facebook was launched

Mark Zuckerberg and his college roommates launched Facebook, initially intended as a social networking platform for Harvard students. Facebook quickly expanded, becoming one of the most influential social media platforms globally.

2011 – IBM's Watson won Jeopardy

IBM's AI supercomputer, Watson, defeated human champions on the game show Jeopardy! This achievement demonstrated the growing power of artificial intelligence in understanding and processing human language.

EVENT CARDS (set C)

1964 – The invention of BASIC programming language

John Kemeny and Thomas Kurtz created the BASIC programming language at Dartmouth College to make programming accessible to students and non-specialists. BASIC became one of the most widely used languages for beginners.

1983 – The Apple Lisa was released

Apple launched the Lisa, one of the first personal computers with a GUI. Although it was not a commercial success, it laid the foundation for the Macintosh and influenced future computer interfaces.

1998 – Google was founded

Larry Page and Sergey Brin founded Google, which began as a search engine project at Stanford University. Google soon became the world's most popular search engine, transforming how people access information online.

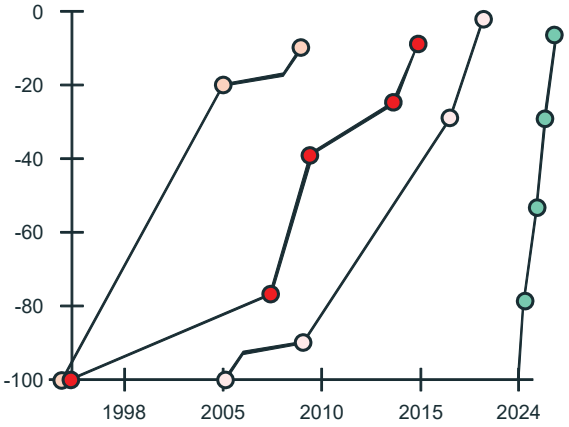
2010 – The iPad was introduced

Apple launched the iPad, a tablet computer that combined features of laptops and smartphones. The iPad opened new possibilities for portable computing and became popular in various fields, from education to entertainment.

2020 – Quantum computing reached a major milestone

Google announced that its quantum computer had achieved "quantum supremacy," completing a calculation in seconds that would take classical supercomputers thousands of years. This event marked a significant step forward in the potential of quantum computing.

Task 14. Study the line chart on test scores of AI systems on various capabilities relative to human performance. Using the expressions learned in the class, interpret data.

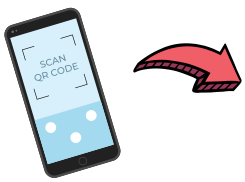


Task 15. Swap the notes with your partner and peer review the data interpretation.

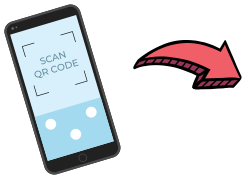
Task 16. Team up and play a word jumble race. The team which unscrambles the word first gets a point for the unscrambled word. Additional point is attributed if the word is appropriately used in a sentence (define the function of an item or a historical figure's role). If a team does not use the word correctly, they get a minus point and another team can try to use the same word in a sentence.

a) NIACE
b) TIRSTASORN
c) BASCAU
d) NIETRNET
e) NUXIL
f) PHIENO
g) NARPAET
h) EADLOEV LCA
i) COSSOLUS
j) SIMACTNOH
k) ORCSOSERP
l) IFTWS
m) OJETS SEVB
n) SEKATZ
o) LIGES SOJAMLG
p) DROIAND
r) TUPMONIACOT
s) VAAJ
t) PRIBE'S
u) NAONES
w) BASCHEMAT RYNEB

Task 17. Play a Bingo game. The teacher will call out words whose antonyms (words of the opposite meaning) you have to find and mark on your Bingo cards. When you match all words, call "Bingo", read aloud the matched words and explain your reasoning.



Task 18. Test your knowledge on the history of computing by scanning the QR code and playing the quiz.



HISTORY OF COMPUTING

Task 1.
1d, 2c, 3f, 4h, 5e

Task 2.
1d, 2e, 3a, 4f, 5h

Task 3.
1) early, 2) browse, 3) from, 4) its, 5) in, 6) enabled, 7) variety

Task 4.

Programming language	Description
FORTRAN	High-level programming languages for scientific and engineering applications.
COBOL	Language for business data processing.
C	Essential for system programming
C++	Extension of C with OOP features.
SQL	Language for database management.
Java	Object-oriented, platform-independent.
JavaScript	Web development scripting language.
Python	High-level, versatile language.
PHP	Server-side scripting language.
Ruby	Dynamic, object-oriented language.
Swift	Apple's language for iOS apps.
Rust	Systems programming with safety features.

Task 5.
1) True. COBOL remains in use, especially in financial and administrative systems, due to its reliability in handling large data processing tasks.
2) False. FORTRAN and COBOL were among the earliest high-level languages. C, developed in 1972, became foundational for system programming.
3) True. Java and JavaScript were both introduced in 1995; Java brought platform independence, while JavaScript allowed for interactive web pages.
4) False. Swift was developed by Apple in 2014 to simplify iOS and macOS development.
5) False. Go, developed by Google, is optimized for large-scale systems and concurrency, making it suitable for cloud computing rather than traditional web development.

Task 6.
1) FORTRAN, scientific 2) C++, classes 3) SQL, databases 4) Python, programming 5) C#, enterprise 6) Rust, memory

Task 7.
Ada Lovelace

Known as one of the first computer programmers, she created an algorithm for Charles Babbage's Analytical Engine.

Alan Turing
Played a pivotal role in developing modern computing and artificial intelligence concepts. He also cracked German codes in WWII using early computing techniques.

Grace Hopper
Developed one of the first compilers and helped pioneer COBOL, a programming language for business applications.

John von Neumann
Developed the von Neumann architecture, which became a foundational model for computer design.

Steve Jobs
Co-founded Apple Inc. and played a major role in popularizing personal computers with user-friendly interfaces.

Bill Gates
Co-founded Microsoft and helped bring personal computing to a global audience through the Windows operating system.

Tim Berners-Lee
Invented the World Wide Web, allowing the internet to become a platform for accessing and sharing information globally.

Linus Torvalds
Created Linux, an open-source operating system that has become widely used in servers and embedded systems.

Mark Zuckerberg
Co-founded Facebook, which became one of the largest social media platforms and contributed to the growth of social networking.

Vint Cerf and Bob Kahn
Known as the "fathers of the internet," Cerf and Kahn developed the TCP/IP protocols that allow different networks to communicate, forming the basis of internet data transmission.
Vocabulary: protocol, TCP/IP, data transmission, networking.

John McCarthy
Coined the term "artificial intelligence" and developed the Lisp programming language, which is still used in AI research.

Seymour Cray
Known as the "father of supercomputing," Cray developed powerful supercomputers that advanced scientific research and computing capabilities.

Niklaus Wirth
Developed the Pascal programming language, designed for teaching programming and structured programming practices.

Guido van Rossum
Created the Python programming language, known for its readability and wide application in data science and web development.

Dennis Ritchie
Co-created the C programming language and was instrumental in developing UNIX, influencing many modern operating systems.

Steve Wozniak
Co-founded Apple Inc. and designed the Apple I and Apple II computers, which contributed to the personal computer revolution.

James Gosling
Created the Java programming language, enabling cross-platform applications and contributing to web development.

Evelyn Boyd Granville
One of the first African-American women in computing, she worked on mathematical calculations for NASA's space missions.

Task 8.
Students' answers

Task 9.
Students' answers

Task 10.
Students' answers

Task 11.
a) In 1928, Alexander Fleming was working in his laboratory at St. Mary's Hospital in London. While he was examining Petri dishes containing colonies of staphylococcal bacteria, he noticed something unusual. One of the dishes had been left uncovered near an open window. A blue-green mold was growing on it, and the bacteria around the mold had disappeared. Fleming realized that the mold, later identified as Penicillium notatum, was killing the bacteria. This unexpected observation led to the discovery of penicillin, the world's first true antibiotic.

b) In 1945, Percy Spencer, an engineer working for the Raytheon company, was experimenting with a magnetron, a device used in radar systems. While he was testing the equipment, he noticed that the chocolate bar in his pocket had melted. Curious, he began placing other foods near the device. When he was observing some popcorn kernels, they suddenly started popping. Spencer realized that microwave energy could cook food quickly. This accidental discovery eventually led to the invention of the microwave oven.

c) In 1956, Wilson Greatbatch, an American engineer, was working on a device to record heart sounds. While he was assembling the circuit, he accidentally inserted the wrong type of resistor. When he turned on the device, it started emitting electrical pulses - very similar to a human heartbeat. Greatbatch realized that this mistake could be used to regulate heart rhythms. Over the next few years, he was developing the idea further, and in 1960, the first successful implantable pacemaker was used in a patient.

Task 12.
Students' answers

Task 13.
Event cards set A)
In 1945, scientists completed ENIAC, the first electronic general-purpose computer, which could perform thousands of calculations per second. While ENIAC was operating, researchers around the world were already thinking of ways to connect computers. Then, in 1969, ARPANET was launched as the first computer network, allowing communication between multiple computers. While ARPANET was expanding, Intel introduced the Intel 4004 microprocessor in 1971, making it possible for computers to become smaller and more powerful. In 1981, IBM released the IBM 5150, one of the first personal computers available to individuals and businesses. At this time, Tim Berners-Lee was developing a new way to link information, which he later called the World Wide Web in 1989. Finally, in 2007, Apple introduced the iPhone, a smartphone that transformed how people communicate, access information, and use technology. This release marked the beginning of the mobile computing era.

Event cards set B
In 1956, IBM developed the IBM 305 RAMAC, the first hard disk drive. It could store 5 megabytes of data, which was revolutionary for its time, even though the device required an entire room. A couple of decades later, in 1973, Xerox introduced the Alto, the first computer with a graphical user interface (GUI). Although it was not commercially released, the Alto's design influenced future personal computers, paving the way for more user-friendly systems. In 1991, Linus Torvalds created Linux, an open-source operating system that encouraged collaboration and revolutionized software development. A decade later, in 2004, Facebook was launched by Mark Zuckerberg and his college roommates. Initially a platform for Harvard students, it quickly expanded, becoming one of the largest social media platforms in the world. In 2011, IBM's AI supercomputer Watson competed on Jeopardy! and won against human champions, showcasing the incredible potential of artificial intelligence in understanding and processing natural language.

Event cards set C
In 1964, John Kemeny and Thomas Kurtz introduced BASIC, a programming language designed to make coding accessible to students and non-specialists. While other programming languages at the time were targeting experts, BASIC became one of the most popular languages for beginners. Almost two decades later, in 1983, Apple released the Lisa, one of the first personal computers with a GUI. Although the Lisa was not commercially successful, it laid the foundation for future Apple products like the Macintosh. In 1998, Larry Page and Sergey Brin founded Google, starting as a university research project. The search engine quickly grew, becoming a key tool for accessing information globally. By 2010, Apple introduced the iPad, a tablet computer that combined the capabilities of laptops and smartphones, opening up new possibilities for portable computing. Most recently, in 2020, Google announced that its quantum computer had achieved quantum supremacy, completing calculations in seconds that would have taken classical supercomputers thousands of years. This achievement marked a significant step forward in the evolution of computing technology.

Task 14.
Students' answers

Task 15.
Students' answers

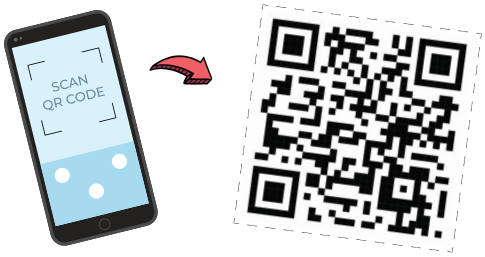
Task 16.

SCRAMBLED WORD	UNSCRAMBLED WORD	CONTEXT
NIACE	ENIAC	The first general-purpose electronic computer, built in 1945.
TIRSTASORN	TRANSISTOR	A small device invented in 1947 that replaced vacuum tubes in computers.
BASCAU	ABACUS	An ancient calculating tool used 4,000 years ago by the Babylonians.
NIETRNET	INTERNET	A global network of interconnected computers.
NUXIL	LINUX	Free open-source operating system based on Unix.
PHIENO	iPHONE	The smartphone that revolutionized mobile technology in 2007.
NARPAET	ARPANET	The precursor to the internet, launched in 1969.
EADLOEV LCA	ADA LOVELACE	The first computer programmer, who worked on Charles Babbage's designs.
COSSOLUS	COLOSSUS	The WWII computer used to decode German messages.
SIMACTNOH	MACINTOSH	Apple's revolutionary personal computer introduced in 1984.
ORCSOSERP	PROCESSOR	The component that performs instructions in a computer.
IFTWS	SWIFT	A programming language developed by Apple, released in 2014.
OJETS SEVB	STEVE JOBS	Co-founder of Apple, known for revolutionizing personal computing.
SEKATZ NOVEWI	STEVE WOZNIAK	Co-founder of Apple, inventor of the Apple I and Apple II computers.
LIGES SOJAMLG	JAMES GOSLING	Creator of the Java programming language.
DROIAND	ANDROID	Google's mobile operating system influenced by Java.
TUPMONIACOT	COMPUTATION	A key concept in programming and software development.
VAAJ	JAVA	A widely used programming language created in 1995.
PRIBE'S NAONES	NAPIER'S BONES	A manual calculation tool invented in 1617, used for multiplication and division.
BASCHEMAT RYNEB	MANCHESTER BABY	The first stored-program computer, built in 1948 at the University of Manchester.

Task 17.

Permanent → Temporary, Analytical → Unsystematic, To store → To throw away, Remote → Nearby, Previous → Future, To adopt → To reject, Milestone → Minor step, Inventor → Follower, To demonstrate → To hide, To manufacture → To destroy, Alongside → Away from, Groundbreaking → Ordinary, Notable → Insignificant, Commercial → Non-profit, Difference → Similarity, Analogue → Digital, Countless → Few, Gradually → Suddenly, Massive → Miniscule, Vast → Small, Widespread → Limited, Accurate → Incorrect, Addition → Subtraction, To incorporate → To exclude

Task 18.



Teacher's notes and lesson plan

Unit overview

This lesson focuses on the history of computing and introduces key milestones, figures, and technological advancements. It includes various skill-building activities such as reading comprehension, listening exercises, discussions, writing tasks, and vocabulary development.

Learning objectives

✓ Identify key milestones in computing history.

✓ Understand and use vocabulary related to computing history.

✓ Improve reading comprehension and ordering of historical events.

✓ Develop listening skills through factual recall and inference.

✓ Enhance speaking abilities through discussion and role-play.

✓ Apply writing skills to summarize and describe events.

Learning outcomes

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

✓ Accurately match historical computing events with descriptions.

✓ Correctly sequence major milestones in computing history.

✓ Listen and comprehend details about programming languages.

✓ Engage in discussions and role-play activities using appropriate terminology.

✓ Write coherent summaries and timeline descriptions.

Lesson plan

Reading

Objectives:

- Develop reading comprehension, identify main ideas, and enhance scanning skills.
- Enhance sequencing skills and reinforce historical context.
- Strengthen understanding of computing terminology and reading comprehension.

Activities:

1) Reading task (Task 1) -15 min
- Students read the text on milestones in computing history.
- They match given titles to corresponding paragraphs.
- Peer discussion to compare answers before class feedback.

2) Ordering events chronologically (Task 2) - 10 min
- Students re-read the text and arrange historical events in order.
- Small group discussions to justify choices.
- Class discussion to confirm the correct sequence.

3) Completing summary with one word (Task 3) - 8 min
- Students fill in blanks in a text summarizing smartphone history.
- Pair work: students compare answers and justify choices.
- Whole-class review.