# Fundamentals of Programming

## Lecture 1

**Chamila Karunatilake**

Department of Information and Communication Technology

Faculty of Technology

University of Sri Jayewardenepura

***chamilakarunatilake@sjp.ac.lk***

# Aim of Module

The aim of this module is to provide theoretical and practical knowledge, understanding, and skills on fundamental concepts of computer programming and cultivate opportunities to utilize them in programming problem-solving.

# Intended Learning Outcomes

On successful completion of this module, students should be able to:

**MLO1** Describe the fundamental concepts of computer programming

**MLO2** Write simple computer programs using programming tools, IDEs, and libraries effectively

**MLO3** Use conventions and best practices of programming

**MLO4** Apply fundamental programming concepts to solve simple problems

# Course Outline

- Introduction to Programming
- Introduction to C programming
- C fundamentals
- Data types
- Operators
- Control structures
- Functions
- Arrays and Strings
- Pointers
- Structures, Unions and Enumerations
- File handling in C

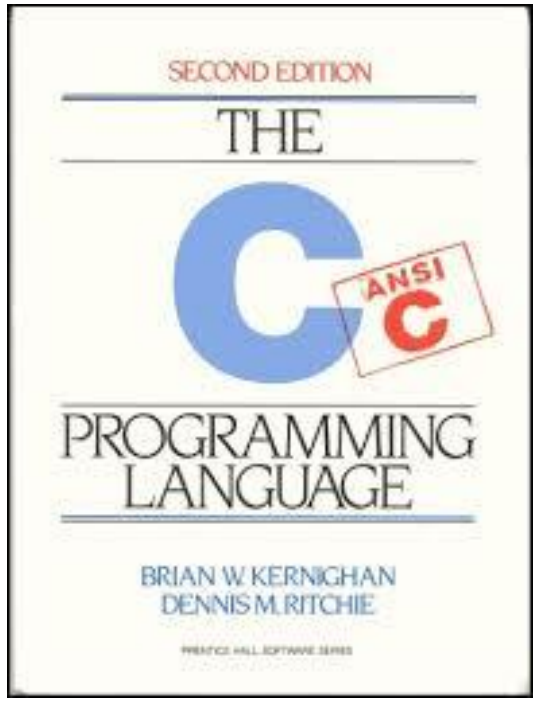
# Assessment

- Continuous Assessments(CA) **30%**
  - Assignments
  - Programming Project
  - Laboratory Exercises
- End Semester Assessment(ESA) **70%**

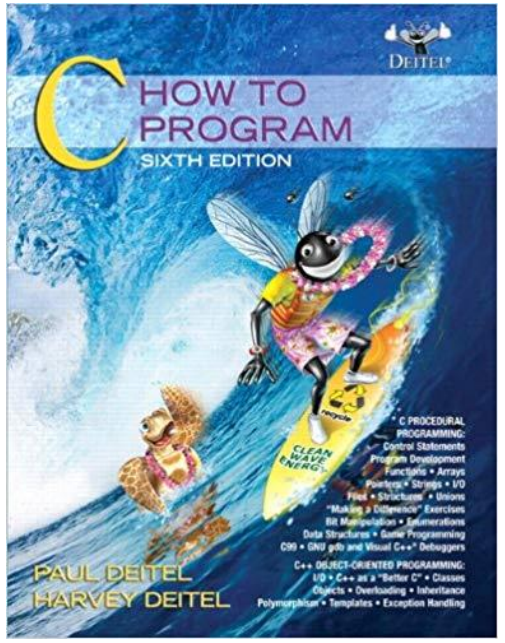
To Pass the exam, CA – 40%

ESE – 35%

# Reference Books




**The C Programming Language**
by Brian W. Kernighan and Dennis M. Ritchie




**C: How to Program**
by Paul Deitel and Harvey M. Deitel

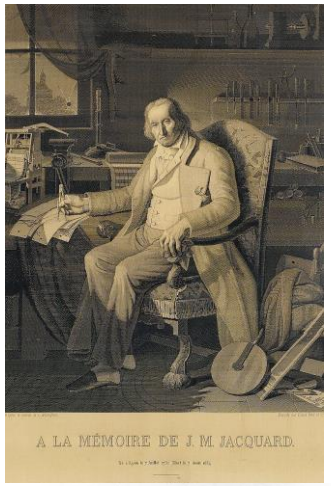
# A brief History of Programming

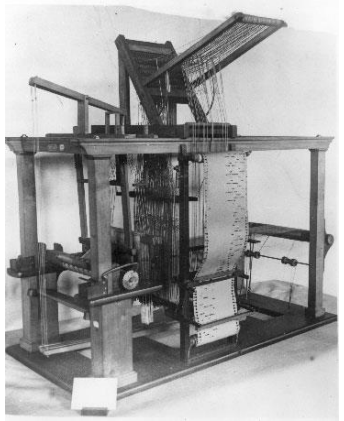
# Hero of Alexandria(10 AD – 70 AD)

- He was a mathematician and engineer who was active in his native city of Alexandria, Roman Egypt.
- Hero invented many mechanisms for the Greek theater, including an entirely mechanical play with puppets which was almost ten minutes in length.
- It was powered by a binary-like system of ropes, knots, and simple machines operated by a rotating cylindrical cogwheel.
- The sound of thunder was produced by the mechanically-timed dropping of metal balls onto a hidden drum.
- These ropes and pulleys could be re-wired in a different way then the puppets of the play would act differently. Therefore, it is considered as a programmable play.

# Joseph Marie Jacquard(1752 – 1834)

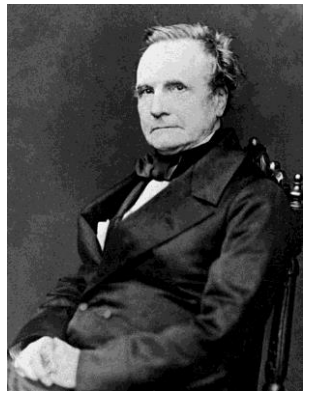
- He was a French weaver and merchant.
- He played an important role in the development of the earliest programmable loom, the "Jacquard loom“.
- It used punched card technique which had metal boards with punched holes. Depending on the binary pattern on the punched cards, the pattern on the fabric is created.
- This mechanism played an important role in the development of other programmable machines, such as an early version of digital compiler used by IBM to develop the modern day computer.

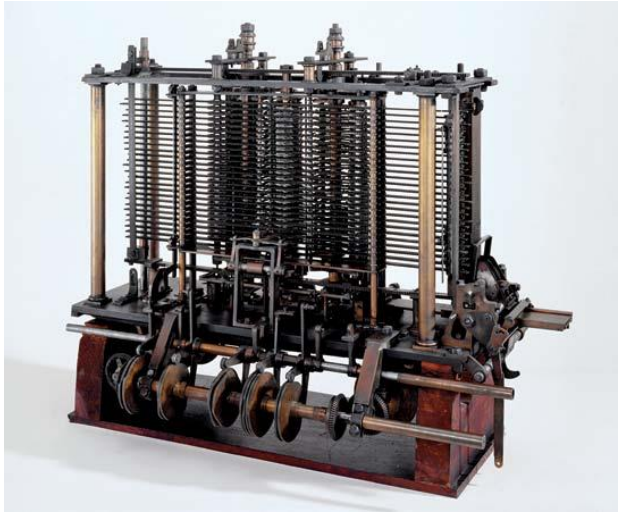


# Charles Babbage(1791 – 1871)

- A British Mathematician who is considered as the **Father of the computer**.
- He designed the first mechanical computer, **Analytical Engine** which was capable of performing mathematical calculations.
- It was never built during his time due to the lack of funds from the British parliament. Later, his design was built into a real machine and was able to work as he predicted.



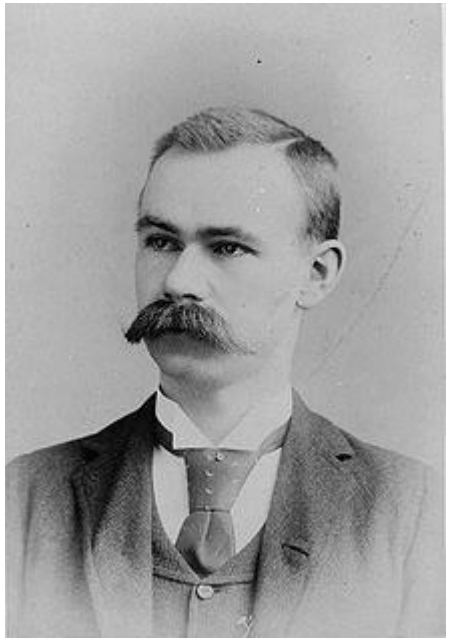


# Ada Lovelace(1815 –1852)

- A British mathematician and she was the daughter of the famous poet Lord Byron and a good friend of Charles Babbage.
- She translated a paper of a young Italian engineer, which was written about Analytical Engine, to English. She added her own notes to the translation and one of those is **an algorithm for the Analytical Engine to compute Bernoulli numbers**. (1842-1843)
- This is considered as the **first computer program;** hence the Ada Lovelace is considered as **the first computer programmer**.
- Later, when analytical engine was built, they checked her program and it worked perfectly!
- She prognosticated the uses of analytical engine is not limited to calculating numbers and it can solve more complex problems such as playing music.

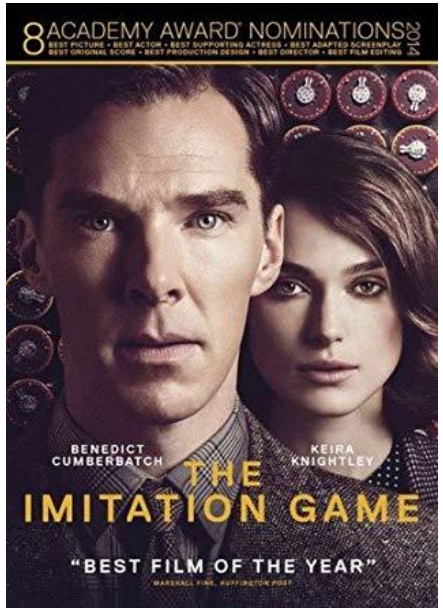
# Herman Hollerith(1860 – 1929)

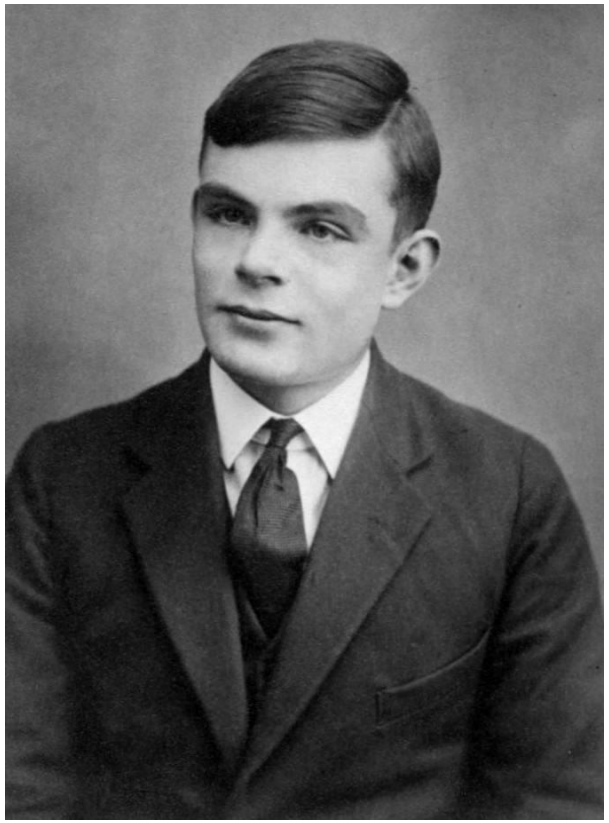
- He was an American inventor who developed the idea of Electric Tabulating System, a machine that could read data.
- He used punched cards which were called Hollerith Cards which form the basis for information processing.
- In 1896 he started the Tabulating Machine Company which later became IBM.
- In 1906, he developed a plug board or control panel, that allows the machine to perform different tasks without being reconstructed.



# Alan Turing(1912 – 1954)

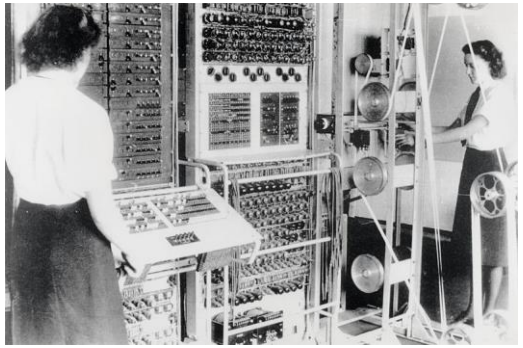
- He was a computer scientist, mathematician and cryptanalyst.
- Turing was highly influential in the development of theoretical computer science, providing a formalization of the concepts of algorithm and computation with the **Turing machine.**



# Tommy Flowers(1905-1998)

- He was a telephone engineer at British post office.
- In 1943, during the world war II, he built **Colossus**, the world's first programmable electronic computer, to help solve encrypted German messages.
- Colossus used vacuum tubes to perform Boolean and counting operations.
- it was programmed by switches and plugs and not by a stored program.
- The existence of the Colossus machines was kept secret until the mid-1970s.

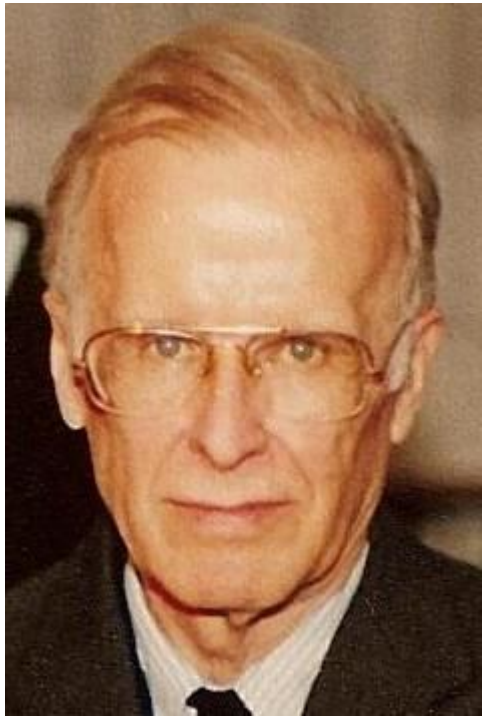


# Konrad Zuse(1910 - 1995)

- He was a  was a German civil engineer, inventor and computer pioneer.
- Konrad Zuse designed Plankalkul (plan calculus), the first algorithmic programming language.

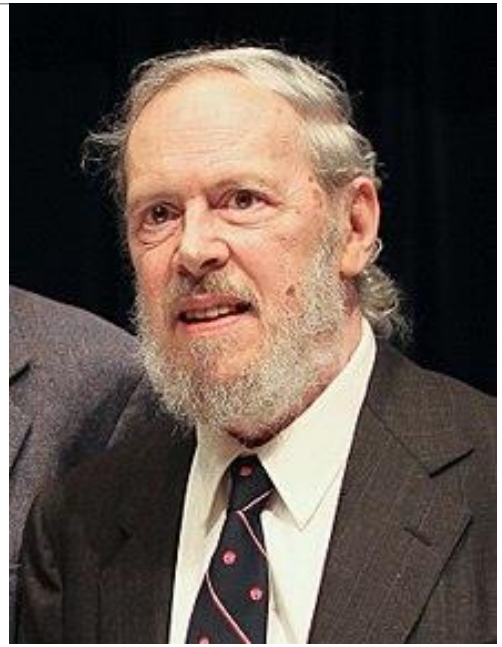
# John Backus(1924 –2007)

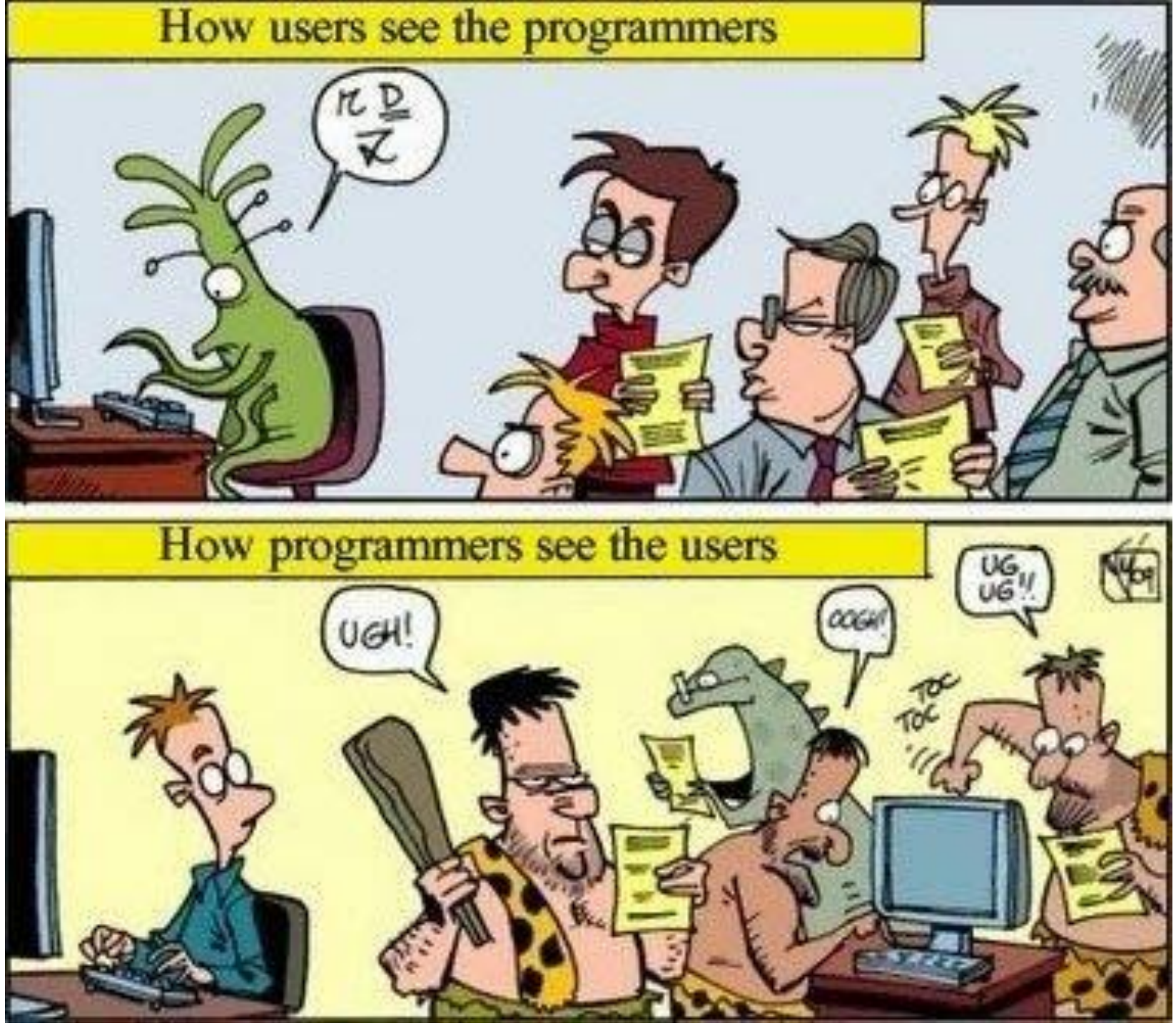
- He was an American computer scientist who worked at IBM.
- He directed the team that invented and implemented **FORTRAN**(FORmula TRANslator), which is considered as the first high-level programming language.
- Also, he invented the Backus–Naur form(**BNF**), a widely used notation to define formal language syntax.
- FORTRAN dramatically changed the way that computers were used.
- Has continued to evolve, adding new features & concepts to the language.



# Dennis Ritchie(1941-2011)

- He was an American computer scientist who worked at Bell Labs Computing Sciences Research Center.
- He created the C programming language in 1971.
- He developed the Unix operating system with Ken Thompson.
- His contribution to the modern advancement of computer science and technology is enormous, for example, iPhone would be a dream without C and UNIX.




How users see the programmers
How programmers see the users
UGH!
OOGH!
UG UG!!
TOC TOC

MAUS

# Questions?