# Tutorial Sheet Solution

## Table of contents



> Warning
>
> **Important**: Your solution does not have to be identical (or even close) to mine, as long as you have a similar line of reasoning, it is more than acceptable.

# 1 Lecture 1: Introduction

- Question 1: Why are computers called general-purpose machines?
- - Solution: Computers are termed "general-purpose machines" because they can handle a broad spectrum of computational tasks, thanks to their programmability. This means that with the right software, a computer can switch from being a word processor to a gaming console, or from analyzing scientific data to playing music. In contrast, "specific-purpose machines" like calculators are designed for a single computational task, such as arithmetic operations. Similarly, a toaster is built specifically for toasting bread and cannot be reprogrammed to, say, boil water. In essence, while a general-purpose machine like a computer can be likened to a multi-tool adaptable to various tasks, specific-purpose machines are more like individual tools, each optimized for a particular function.
- **Question 2**: Identify some advantages and disadvantages of a using a Biometric authentication system in a company (Hint: think about privacy)?
- - Solution:

| Advantages | Disadvantages |
|---|---|
| **Strong security:** Biometric data is unique to each individual, making it difficult to fake or replicate. This can help to prevent unauthorized access to systems and data. | **Privacy concerns:** Biometric data is considered to be sensitive personal information. There are concerns that biometric data could be used to track people or discriminate against them. |
| **Convenience:** Biometric authentication can be more convenient than traditional methods, such as passwords and PINs. Users do not have to remember or carry anything, and the process can be quick and easy. | **Irrevocability:** Biometric data cannot be changed like passwords. If someone's biometric data is compromised, they cannot change it, which could lead to security vulnerabilities. |
| **Accountability:** Biometric systems can provide a clear audit trail of who accessed what and when. This can help to track activities and ensure individual accountability. | **False positives/negatives:** No biometric system is 100% accurate. There is always a chance of false rejections (an authorized person being denied access) or false acceptances (an unauthorized person gaining access). |

- **Question 3**: Is Computer Science an Engineering discipline? Explain why or why not?
- - Solution: Computer Science merges the theoretical underpinnings of computation with their practical applications, making it a blend of science and engineering. At its heart, it delves into scientific aspects like algorithms, data structures, and the boundaries of computation. Yet, it also embodies engineering when applying these theories to real-world challenges, as seen in software engineering, computer architecture, and networking. This dual nature defines Computer Science, emphasizing its role in understanding computational principles and addressing real-world needs. In essence, Computer Science is not exclusively an engineering discipline but a hybrid of engineering and science as we talk about both the theory and its application in CS.

If you would like to read more about what is CS, I recommend you read Rapaport, William J. "What is a computer? A survey." Minds and machines 28 (2018): 385-426.

- **Question 4**: Why do we need standardization in computer Science?

Standardization in computer science ensures that systems and software operate consistently and efficiently across different platforms and environments. It facilitates interoperability, meaning different systems can work together seamlessly.

For example, consider HTML, the language used to create web pages. Without a standardized way to write and interpret HTML, every web browser (like Chrome, Firefox, or Safari) might interpret the same code differently. This could lead to a website looking perfect in one browser but broken in another.

Simple Example:

Imagine you wrote the following HTML code to display a heading:

```
\<heading\>BCS1110\</heading\>
```

If HTML wasn't standardized, one browser might display "BCS1110" as a large, bold title, while another might not recognize the `<heading>` tag at all and just display the text without any special formatting.

However, with standardization, we use the recognized tag `<h1>` for headings:

```
<h1>BCS1110</h1>
```

All browsers that adhere to the HTML standard will display this consistently as a primary heading.

In conclusion, standardization in computer science, as illustrated with HTML, ensures consistency, reliability, and interoperability across various systems and platforms.

- **Question 5**: A recipe database stores the recipes in Table. Users can search for recipes by entering search terms, which the database matches to tags and cooking times. Consider the following search terms and decide which recipes will be returned:

```
-   A. cooking time less than 20 minutes and not vegetarian;

-   B. includes chicken or turkey but not garlic;

-   C. doesn't include nuts.
```

| Name | Tags | Cooking time |
|---|---|---|
| Broiled chicken salad | Chicken, lettuce, gorgonzola cheese, lemon juice | 15 mins |
| Holiday turkey | Turkey, rice, onion, walnuts, garlic | 60 mins |
| Three-spice chicken | Chicken, ginger, cinnamon, garlic, green beans | 30 mins |
| Lentil salad | Lentils, onion, peppers, walnuts, lettuce | 20 mins |
| Garlic dip | Garlic, lemon juice, chickpeas, chicken broth | 5 mins |

- 
    - Solution:

```
- A. cooking time less than 20 minutes and not vegetarian: Garlic dip;
- B. includes chicken or turkey but not garlic: Broiled chicken salad;
- C. doesn't include nuts: Broiled chicken salad, Three-spice chicken.
```

- **Question 6**: Here are some rules of thumb to apply when deciding which supermarket queue to join (assuming you want to spend as little time queuing as possible)
  - A. People in a queue who are carrying their items by hand take about one minute to be processed.
  - B. People carrying a basket take about two minutes to be processed.
  - C. People pushing a trolley take about five minutes to be processed for a half-empty trolley; ten minutes for a full trolley.
  - D. People in a self-service checkout are processed in about 80 per cent of the time of a normal checkout.

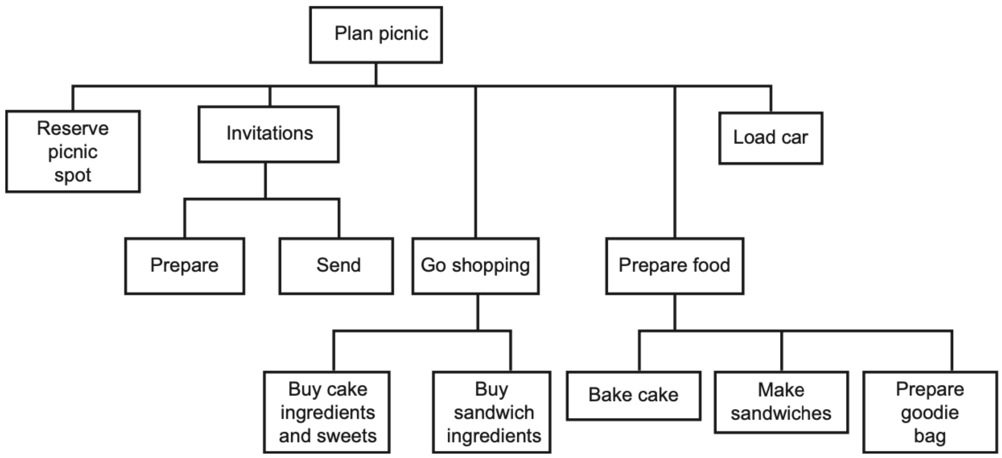
    Express these rules of thumb as logical statements. Each statement should make a conclusion about the estimated queuing time.
- - Solution:
    * A. if customer is carrying items by hand, then time is 1 minute;
    * B. if customer is carrying a basket, then time is 2 minutes;
    * C. if customer is pushing a trolley, then:
      · i. if trolley is half-full, then time is 5 minutes;
      · ii. if trolley is full, then time is 10 minutes.
    * D. if customer is at a self-service checkout, reduce time by 80 per cent.

**Note**: Any logical statement is acceptable here, some of you have come up with pseudocode while others have written some code, all of it is acceptable as long as it has the if statement in it.

- **Question 7**: You're planning to hold a birthday picnic for a child and her friends. Break down the preparation into a tree structure of tasks. The facts are:
  - A. You need to send out the invitations to the parents of the other children.
  - B. Food you'll provide: sandwiches (ham, chicken, and cheese), homemade cake.
  - C. Fresh ingredients (meat and dairy) need to purchased on the day.
  - D. Other things you need to take: disposable cutlery, blankets, games.
  - E. The park where the picnic is held requires you to reserve a spot.

- F. All guests will get a goody bag with sweets when they leave.

- - Solution:




- **Question 8**: Order the following into layers of abstraction, starting with the most general and ending with the most specific:
  - A. penguin;
  - B. bird;
  - C. Tommy the Penguin;
  - D. animal;
  - E. Emperor penguin.

- - Solution: animal, bird, penguin, Emperor penguin, Tommy the Penguin.

## 2 Lecture 2: Hardware

1. Applying Moore's Law, how much larger would you expect a computer from 30 years ago to be by comparison to the computer you currently use?

- Solution: Moore's Law, proposed by Gordon Moore in 1965, posits that the number of transistors on a microchip doubles approximately every two years, which often translates to a doubling of computational power.

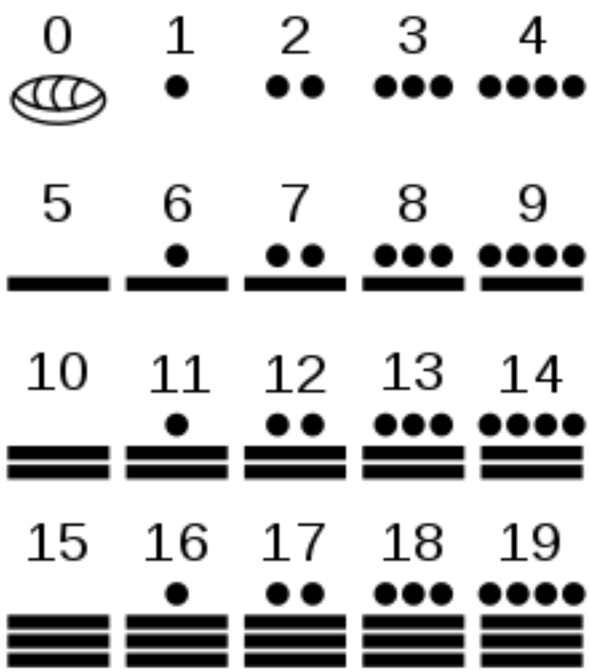
  To calculate the factor by which a computer's power has increased over 30 years based on Moore's Law:

  1. Determine the number of two-year periods in 30 years: ($\frac{30 \text{ years}}{2 \text{ years/period}} = 15$ periods)
  2. Since the power doubles every period, the increase is ($2^{15}$).

  ($2^{15}$) = 32,768

  So, based on Moore's Law, a computer from 30 years ago would be approximately 32,768 times larger than a modern computer, assuming that the law held perfectly true over that time span.

2. Using the Mayan numeral system, what are the decimal numbers 12, 123, 452, and 1256? Research online for your answer.

- Solution: 

In the Mayan system, numbers beyond 19 are depicted vertically using a base-20 format. This is analogous to the Hindu-Arabic system which operates on a base-10 principle. To illustrate, the number 33 is portrayed with a single dot over a combination of three dots and two bars. Here, the solitary dot signifies “1×20”, and when combined with the value represented by the three dots and two bars (which is 13), the total is 33. When numbers reach 202 or 400, a new row is initiated, with sequences like 203, 8000, 204, 160,000, and so forth. For the number 429, it’s expressed as a dot over another dot, followed by four dots and a bar, translating mathematically to (1×202) + (1×201) + 9 = 429.

**12**: (2 x 5 + 2) x 20^0 = 12 


**123**: (1 x 5 + 1) x 20^1 + 3 x 20^0 = 123


**452**: 1 x 20^2 + 2 x 20^1 + (2 x 5 + 2) x 20^0 = 452


**1256**: 3 x 20^2 + 2 x 20^1 + (3 x 5 + 1) x 20^0 = 1256

> ⚠ Warning
>
> **Don’t worry**: You do not need to memorize anything here. I will not ask you questions in exam without providing your all the information you need to covert numbers from one system to another (replacing the bit in this question where you were expected to google, you still need to understand the conversion process and logic)

3. Each text message contains a date (8 bytes), time sent (4 bytes), up to 160 characters of text, and the sender’s phone number (8-byte number). How many text messages can you store on a 128GB hard drive? [*Hint: Read the required reading for this lecture*]

- Solution:

  Each text message will need a date (8 bytes), a time sent (4 bytes), text (up to 160 characters), and a phone number (8 bytes). If we assume each character is stored in ASCII representation, then each character will take 1 byte. So each text will need 8 + 4 + 160 + 8 = 180 bytes. 128GB = 128,000MB = 128,000,000KB = 128,000,000,000Bytes. So we could fit 128,000,000,000 Bytes / 180 Bytes per text. That gives 711,111,111 texts.

4. What would happen if you ran out of RAM? What would happen if you ran out of permanent storage?

- Solution:

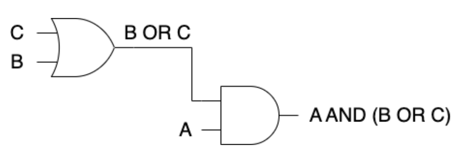
  RAM is the computer's temporary workspace. If you ran out of RAM, your programs wouldn't be able to run. Most likely your computer would slow down tremendously if you ran out of RAM.If you ran out of permanent storage, your computer would no longer be able to permanently store any of its files. You would no longer be able to save any work.

5. Complete the truth table for the logical statement A AND (B OR C) and draw the circuit diagram for this system.

- Solution:

  The truth table for the logical statement A AND (B OR C):

| A | B | C | B OR C | A AND (B OR C) |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |




6. Show how to convert the following binary numbers to decimal:

   a. 1101
   b. 1111101000

c. 11.0011
d. 11.110001

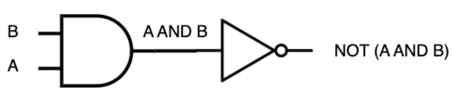
- Solution:
  - a. 1101: $(1101)_2 = (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2\ ) = (13)_{10}$
  - b. 1111101000: $(1111101000)_2 = (1 \times 2\ ) + (1 \times 2\ ) + (1 \times 2\ ) + (1 \times 2\ ) + (1 \times 2\ ) + (0 \times 2\ ) + (1 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (0 \times 2\ ) = (1000)_{10}$
  - c. 11.0011: $(11.0011)_2 = (1 \times 2^1) + (1 \times 2\ ) + (0 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3}) + (1 \times 2^{-4}) = (3.1875)_{10}$
  - d. 11.110001: $(11.110001)_2 = (1 \times 2^1) + (1 \times 2\ ) + (1 \times 2^{-1}) + (1 \times 2^{-2}) + (0 \times 2^{-3}) + (0 \times 2^{-4}) + (0 \times 2^{-5}) + (1 \times 2^{-6}) = (3.765625)_{10}$

7. Design truth tables and ciruit diagrams for the following Boolean expressions:

   a. not (a and b)
   b. not (a or b)
   c. not a or not b
   d. not a and not b

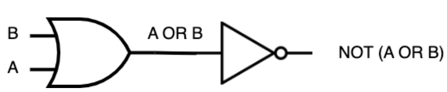
Let's start by creating the truth tables for each of the given Boolean expressions:

**a. not (a and b)**




| a | b | not (a and b) |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

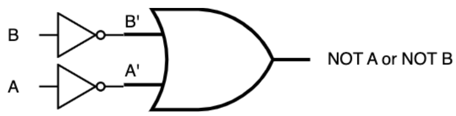
**b. not (a or b)**




| a | b | not (a or b) |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

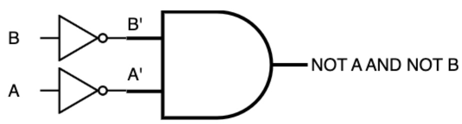
| a | b | not (a or b) |
|---|---|---|

**c. not a or not b**




| a | b | not a | not b | not a or not b |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |

**d. not a and not b**




| a | b | not a | not b | not a and not b |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |