Unit 5.3 Highlights

Computational Thinking (CT) is the thought process involved in formulating a problem and expressing its solution(s) in such a way that a computer—human or machine—can effectively carry out. Three steps for CT:
Abstraction: Problem formulation
Automation: Solution expression
Analyses: Solution execution and evaluation
(Source: Wikipedia)

— —

JavaScript String Methods (Source: W3Schools)

charAt() Returns the character at the specified index (position)
charCodeAt() Returns the Unicode of the character at the specified index
concat() Joins two or more strings, and returns a new joined strings
endsWith() Checks whether a string ends with specified string/characters
fromCharCode() Converts Unicode values to characters
includes() Checks whether a string contains the specified string/characters
indexOf() Returns the position of the first found occurrence of a specified value in a string
lastIndexOf() Returns the position of the last found occurrence of a specified value in a string
localeCompare() Compares two strings in the current locale
match() Searches a string for a match against a regular expression, and returns the matches
repeat() Returns a new string with a specified number of copies of an existing string
replace() Searches a string for a specified value, or a regular expression, and returns a new string where the **specified** values are replaced
search() Searches a string for a specified value, or regular expression, and returns the position of the match
slice() Extracts a part of a string and returns a new string
split() Splits a string into an array of substrings
startsWith() Checks whether a string begins with specified characters
substr() Extracts the characters from a string, beginning at a specified start position, and through the specified number of character
substring() Extracts the characters from a string, between two specified indices
toLocaleLowerCase() Converts a string to lowercase letters, according to the host's locale
toLocaleUpperCase() Converts a string to uppercase letters, according to the host's locale
toLowerCase() Converts a string to lowercase letters
toString() Returns the value of a String object
toUpperCase() Converts a string to uppercase letters
trim() Removes whitespace from both ends of a string
valueOf() Returns the primitive value of a String object

— —

Algorithms (Source: Essential Algorithms, Rod Stephens, Wiley 2013)

Algorithm - Recipe for performing a certain task
Data structure - Arrangement of data to make solving a particular problem easier
Pseudocode - Text which gives an idea the structure and details needed to implement the algorithm in code *without* using programming language syntax

Considerations of an algorithm
Behavior. Does it find the best possible solution, or does it just find a good solution? Could there be multiple best solutions? Is there a reason to pick one "best" solution over the others?

Speed. Is it fast? Slow? Is it usually fast but sometimes slow for certain inputs?

Memory requirements. How much memory will the algorithm need? Is this a reasonable amount? Does the algorithm require billions of terabytes more memory than a computer could possibly have (at least today)?

Techniques. Can you reuse those techniques to solve similar problems?