Algorithms and Computer Programs

Lesson Objectives

- Know basic computer operations
- Understand algorithms and it's parallel to computer programs
- Learn to declare and assign value to a variable in JavaScript.

Computer Operations

- Input (Receive Data)
  - Input devices: keyboard, mouse, camera, mic, sensors
- Process (Execute instructions, usually to operate on data)
  - You write **algorithms** and computer programs for this.
  - Computer can only process data/instructions stored in memory.
- Output (Present computed result)
  - Output devices: monitor, speakers, printers

- You can compare the computer operations with a function in Mathematics, it also has input, process and output.
  - Your task is to write the function definition when you know the input and the expected output of a function.

Algorithms

- Sequence of instructions to achieve a specific task
  - Recipes to prepare a meal
  - Driving instructions

- Algorithm to brush teeth
  - Hold your toothbrush
  - Pour small amount of toothpaste
  - Start brushing
  - Stop brushing after 2 mins or when you feel fresh enough.
  - Rinse your mouth

Example

- Algorithm for computing the average test score from a list of test scores.
  1. Set *sum* to 0
  2. Get the number of Students
  3. For each student
     a) Get the student's test score
     b) Add the student's test score to sum
  4. Divide the *sum* by the number of students

- Notice that it is nothing more than the step-by-step process for performing the calculation.

Computer Program

- Algorithm that a machine can execute
- A program is composed of data and instructions.
  - Data (like ingredients of a recipe)
    - Test scores in the prior example
  - Instructions (like steps to prepare a recipe)
    - Step by step process to compute average score in the prior example.

Machine is Dumb

- The computer doesn't know anything
  - About anything at all

- The computer (memory) can store things
  - Data items that it is told to store (e.g., test score for a student)
  - Algorithms/Instructions (steps) aka programs (e.g., compute average)

- The computer (processor) can read and execute the instruction saved in memory to access/manipulate data in memory.

Instructions

- The instructions in a computer program are like the steps in algorithm.
  - In programming language like JavaScript, the instructions are called statements.
  - Statements are syntax constructs and commands that perform actions.

- Instructions access and update data
  - The program to calculate average test score would get (access) test score from every student.
  - Would update the value of *sum* after adding test score for each student.

Main point 1

- An algorithm is a step-by-step sequence of operations to accomplish some task. A computer program is an algorithm written using a language that the machine understands. *Science of Consciousness,* The source of thought is pure silence, and thought is the source of all languages. *We experience the field of pure silence during our practice of the Transcendental Meditation Program*.

Hello, world! (the very first statement)

- A program is composed of statements.
  - Statements are syntax constructs and commands that perform actions.
  - Statements can be separated, end with a semicolon (optional).
  - A single statement is equivalent to a single step in an algorithm.
  - A statement is a complete command within a program.

```
console.log("Hello, world!");
```

- What a about the data?
  - What can be considered as data in this statement?

Storing data in a variable

- In computer programs, variables are used to store data or information.
  - A variable is a “named storage” for data.
  - To create a variable in JavaScript, use the const or `let` keywords

```
// declaring or creating a variable
let message;

// assigning some value to a variable
// note we are not using keyword let this time
message = "Hello, world!"

// using data (variable value) in our statement
console.log(message);
```

```
// single line comment
/*multi-line comment */
```

In older scripts, you may also find another keyword: **var** instead of **const** or **let**

Comments

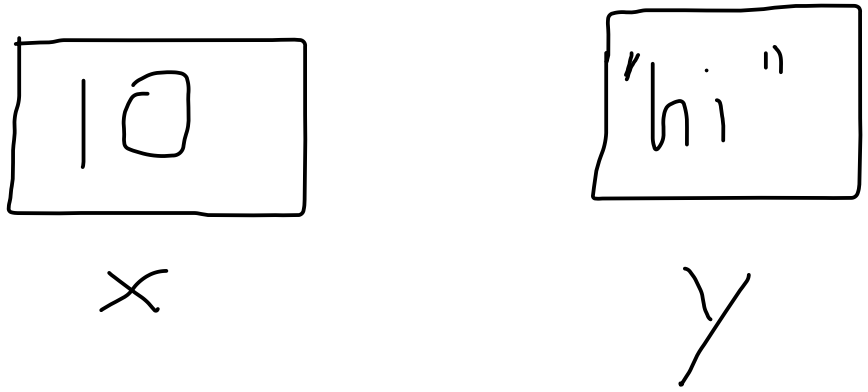
- A comment in a programming language is note or documentation that a programmer writes for himself (future reference) or other programmers who will use the code.
- Comments are ignored during code execution.
- There are two styles of commenting in JavaScript:
  - Single line comments, starts with //
  - Multiline comments, starts with /* and ends with */

Variables

- Memory location referenced by some identifier like x and y.
  - Machine saves data on those memory locations.
  - Machine access/ manipulates variables in a program in order to compute results.

  const x = 10;

  const y = “hi”;



Variable naming

- There are two limitations on variable names in JavaScript
  - The name must contain only letters, digits, or the symbols $ and _
  - The first character must not be a digit.

```
// valid
let username, test123;

// invalid
let 2ndName, my-name;
```

- There is a list of reserved words, which cannot be used as variable names because they are used by the language itself.

- *Case sensitive*
- *Always write meaningful variable names.*

Assigning values to a variable

- When variables are declared, their default value is 'undefined’

```
let total;
console.log(total) //
undefined

total = 5;
console.log(total) // 5
total = total + 10;

console.log(total); // 15
```

Concise ways of assigning variables

```
// define a variable and assign the value
let message = 'Hello!';
alert(message);

// multiple variables in one line
let name = 'John', age = 25, message = 'Hello';

// The multiline variant is bit longer, but easier to read, favor this
let name = 'John';
let age = 25;
let message = 'Hello';
```

Note: use of quotes for text vs numbers

The equals sign

- Many programming languages use the equals sign for assignment (storage) into variables.
- Important! The equal sign in most programming language does not indicate equality.
  - It indicates assignment

Exercise

- What will be the output of the following

```
let x = 5;
let y = 10;
let z = x;

x = y;

console.log(x,y,z);

// how many variables do we have in total?
```

Note: think of a variable as a box or container that can store one value at a time.

Declaring & assigning constants

- To declare a constant (unchanging) variable, use `const` instead of `let`

```
const WEEK_DAYS = 7;
const PI = 22/7;
```

- Convention is upper case for constants known in advance
  - can be used throughout program
  - easy to update in single place
  - Lower case for (local) const variables that you do not expect the program to change

const name = friendList[0];

The modern mode, “use strict”

- The directive looks like a string: "use strict“. When it is located at the top of a script, the whole script works the “modern” way. (Reading)

- E.g., using variables without declaring is a source of many bugs
  - E.g., typo: let specialBankBalance … → specailBankBalance = 1000; …
  - Strict mode reports error

Good coding practice

- Use `const` if the value won't change after assignment
- Use `let` for variables that need to be reassigned
- favor const

- **Never use `var`**

Demo

- Write code to output 5+5
- Update code to save some integer values on **constants** X and Y and print the sum to the console (console.log).
- Update code to save some integer values on re-assignable **variables** x and y and print the sum.

Lab

- We will do the environment setup together
- Download and install VSCode
- Download and install NodeJS
  - Try to run some JavaScript code on Node.js REPL
- Open account at github.com and create a repository named cs301
- Download and install GitHub desktop and clone repository you created in step1 to your machine.
  - File > options to sign in to GitHub.com account
  - File > clone repository
- Open in your external editor (select Visual Studio Code)
  - Create a folder, test, and then a JavaScript file and enter some text
- Commit the changes to master and then Publish branch (back in GitHub Desktop)
  - View on GitHub.com

Assignments

- Read contents from referenced links.
  - What happens when an assignment is performed without using `let or const` keywords in strict mode?
  - List 5 reserved keywords in JavaScript.

- How is a computer program similar/ different than an algorithm.
- Write at least two algorithms you perform on your daily life, with at least one algorithm that can be converted to computer program.
- Write the demo code from today's lesson in VSCode
  - Put it in a file, introDemoCode.js, in a folder, w1d1algorithms
  - Commit it to your 301 repository and push it to GitHub
  - Also submit your answers to the questions above and the two algorithms as a text file to your GitHub account
- Submit your status report for the day as per the instructions in Resources > assignments

References

- The modern mode, "use strict" (javascript.info)
- Variables (javascript.info)