# Programming Workshop 2017

Day 1: Basic concepts in programming

# What is a computer program?

Audience’s answers:

-

# What is a computer program?

A **detailed** set of instructions given to the computer that allows it to perform a certain task.

Examples of computer programs:

1. The mean/std function in excel that calculates mean and standard deviation of experiment data. (simple)
2. Firefox or chrome that allow us to browser the internet. (complex)
3. Games (like Witcher 3 / NFS etc) which calculate how the fictitious world should look/behave based on the user input.

# What is an algorithm?

Audience’s answers:

-

# What is an algorithm?

An algorithm is a set of **high level steps** to solve a given problem.

Examples of algorithms:

1. Navigation: Instructions on how to reach home from NCBS.
2. Cooking: Recipes are detailed algorithms on how to start from raw ingredients and obtain (delicious) food.
3. Protocols (the good ones)

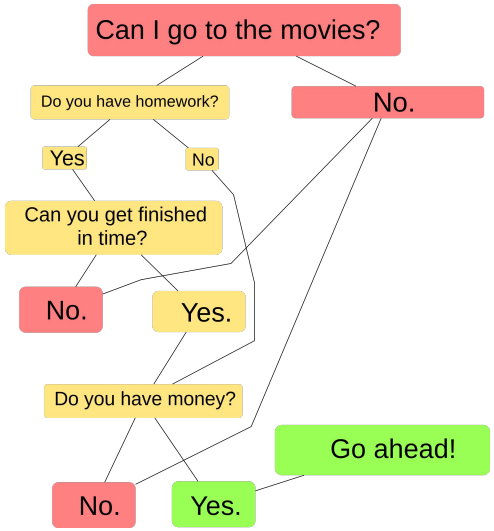
etc etc…

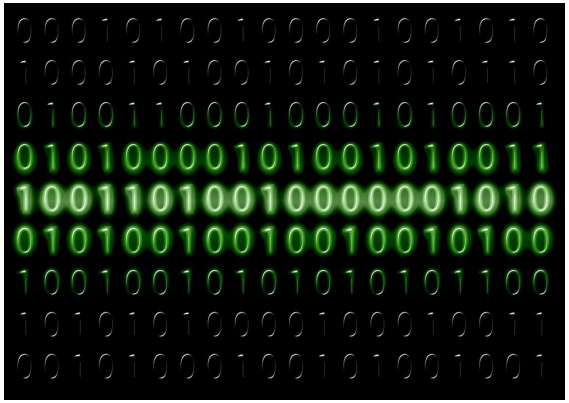
# Science (or art) of programming

Algorithm

Programming languages

Computer Program



Simple enough. Why do people struggle with it then?
Two main reasons:

1. Bad algorithms
2. Bad understanding of how computers work

# Science (or art) of programming

Algorithm

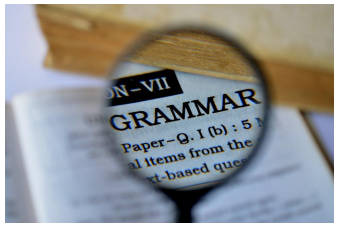
Solution to our problem of interest

Requires: Logic

(simple, but requires practice)

General understanding of Computers

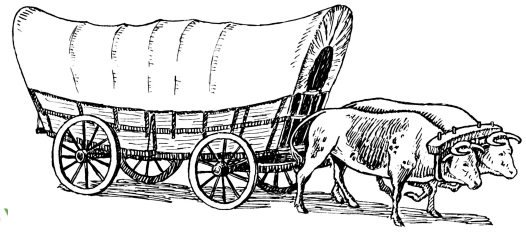
The grammar of programming languages (easy to learn)




Programming Languages (Python, MATLAB, R)

Vocabulary (syntax)

(requires practice, to become an expert)



# Aim of the programming workshop

Tackle the common problems faced:

1. Help out with algorithmic thinking.
2. Give a general understanding of how computers work.
3. Help out during the initial vocabulary learning phase of a language.

# Find the largest number in a list

- Write an algorithm to find the largest number in a list of positive numbers.
- Here are some example sets to try and work on:
  a. [1 3 7 4 3 4 3 3 7 3]
  b. [13 19 9 7 5 18 10 8 17 11]
  c. [49 46 63 62 25 14 8 62 35 36]
  d. [630 260 298 538 176 596 23 693 791 271]
  e. [4445 4308 9309 9334 6861 6005 8350 3712 6567 5264]
  f. [841238 619645 209407 435618 832714 887185 164080 939827 179879 517613]

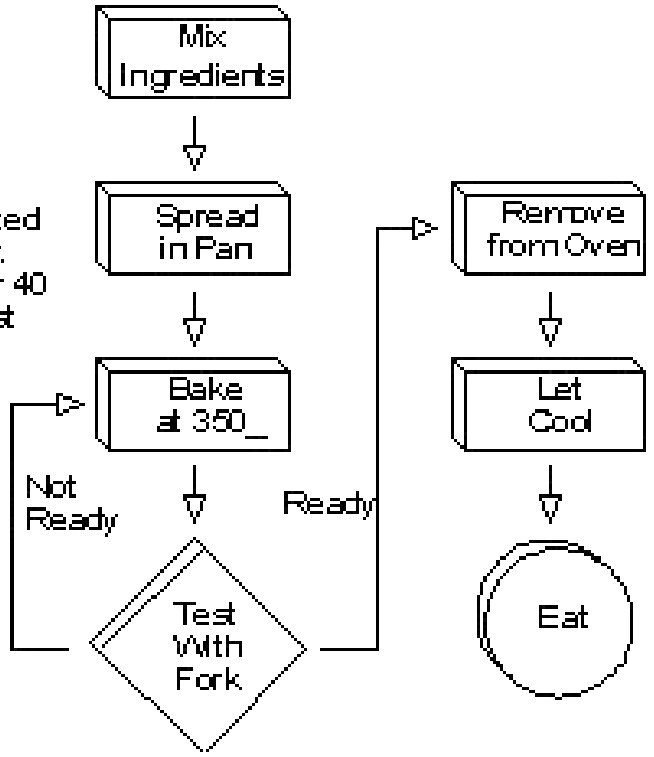
- Audience's answers

# Drawing out the algorithm as a flow chart

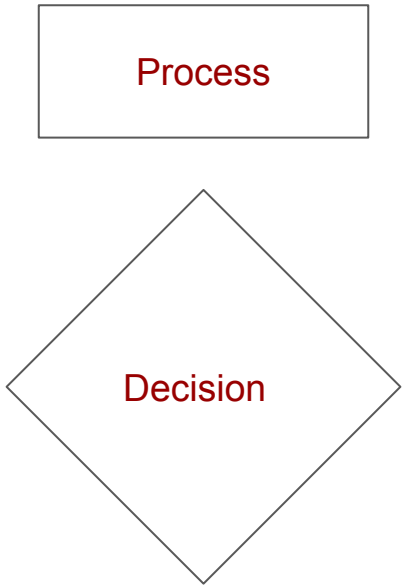
Recipe
CHOCOLATE CAKE

| | |
|---|---|
| 4 oz. chocolate | 3 eggs |
| 1 cup butter | 1 tsp. vanilla |
| 2 cups sugar | 1 cup flour |

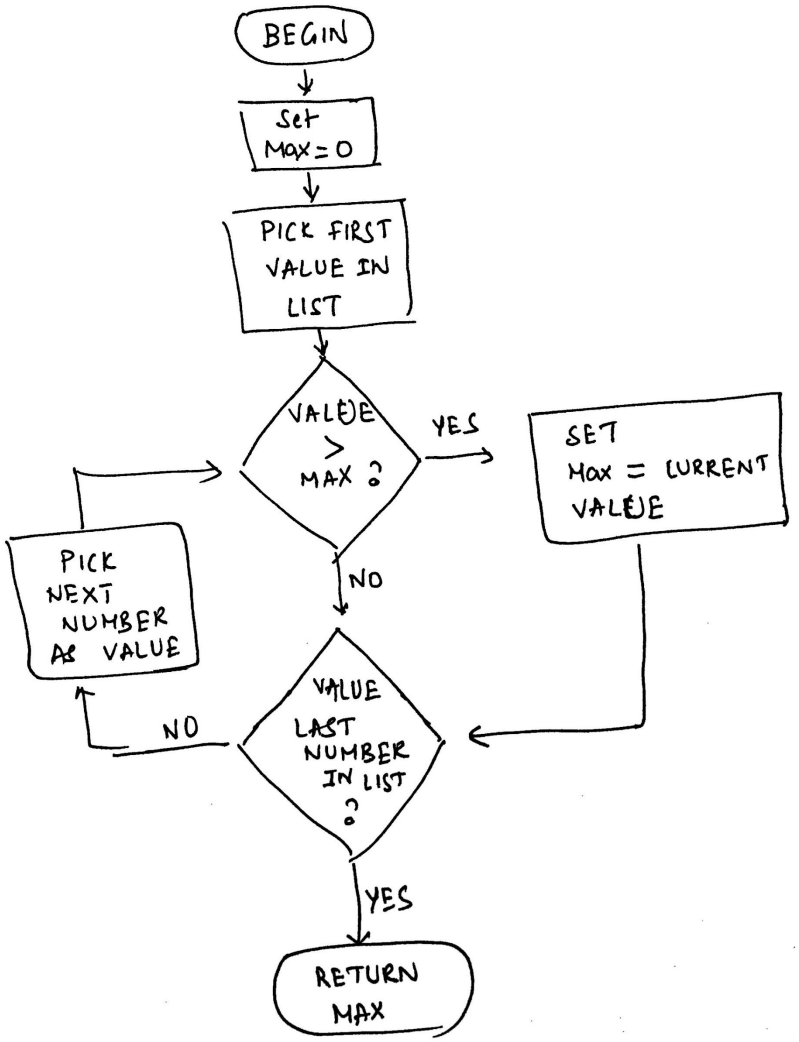
Melt chocolate and butter. Stir sugar into melted chocolate. Stir in eggs and vanilla. Mix in flour. Spread mix in greased pan. Bake at 350_ for 40 minutes or until inserted fork comes out almost clean. Cool in pan before eating.



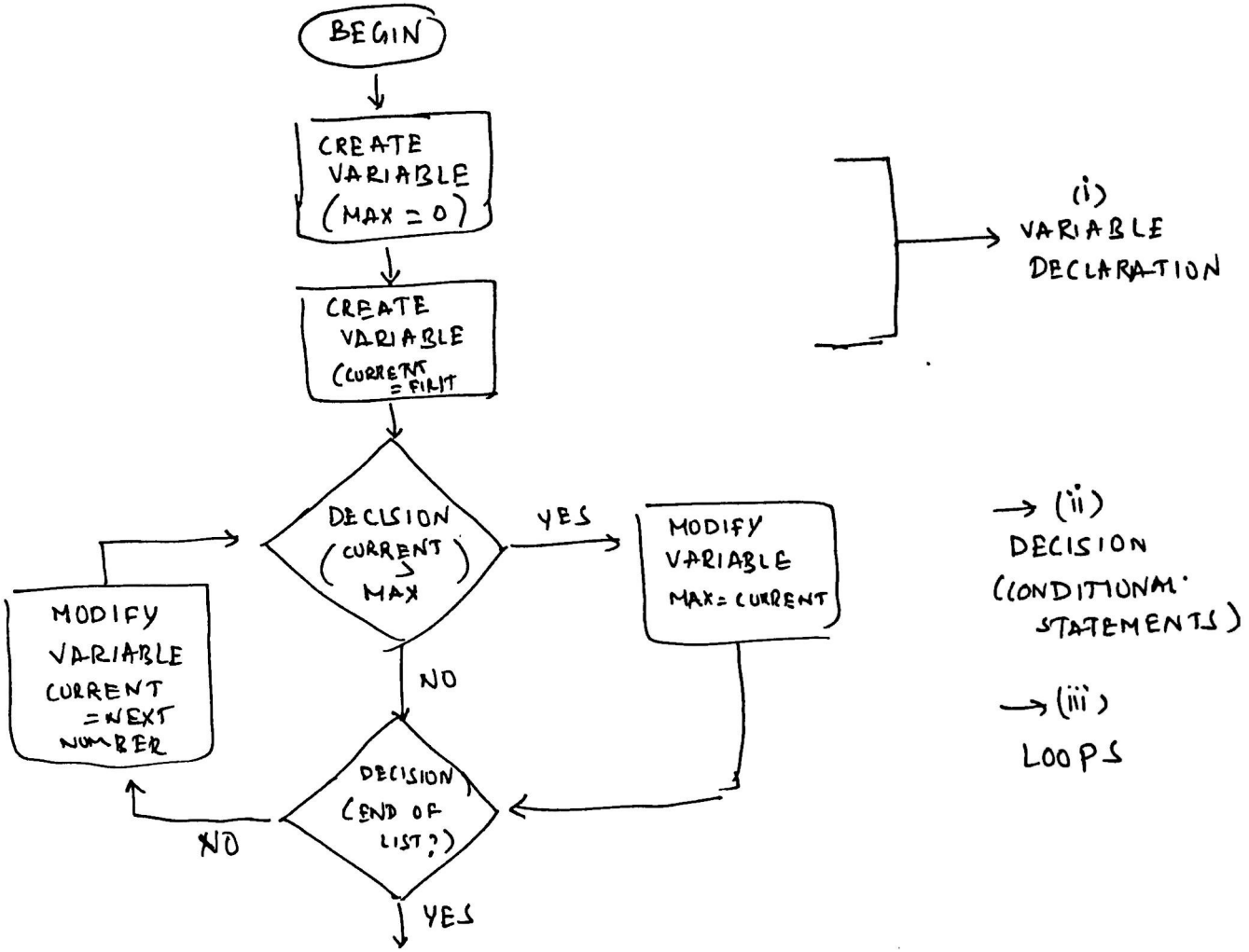

**Lego blocks for this exercise**



# Max number algorithm as a flow chart



# Backbone of max number algorithm



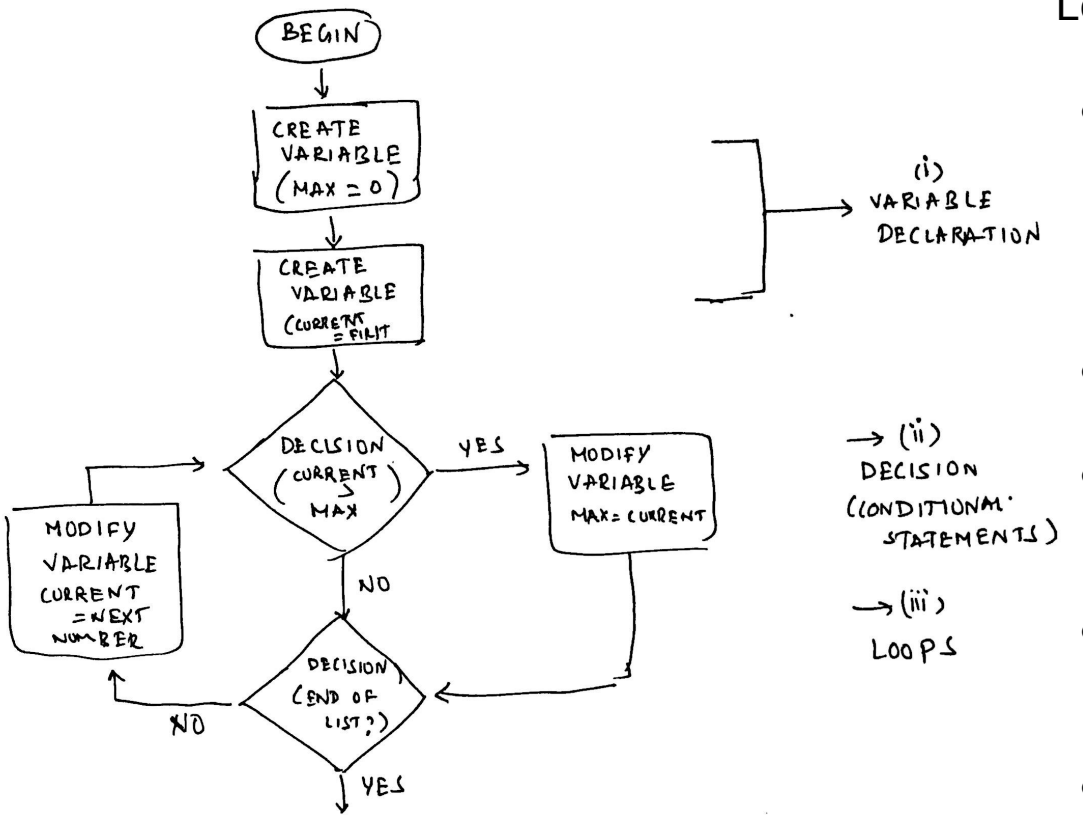
# Variable types

- Basic variables (Grammar)
  - Integer (3)
  - Decimals (example pi) with float type giving ~7 digits and double giving ~16 significant digits after decimal point.
  - Character (‘a’)
  - String - (‘code!’)
  - Boolean - true or false

- Data structures (Vocabulary)
  - Arrays (vectors) - list of decimals (or integers)
  - Lists (python) / cells (matlab) / - an array with any one or all of the basic variables
  - Matrixes - 2 (n) dimensional list of numbers
  - Dictionary (python) - arrays with keys as indices
  - Table / data frame - Columned table with column containing one type of variable
  - etc etc…

# Examples (and cautionary tales)

# Hands on: Variable declaration



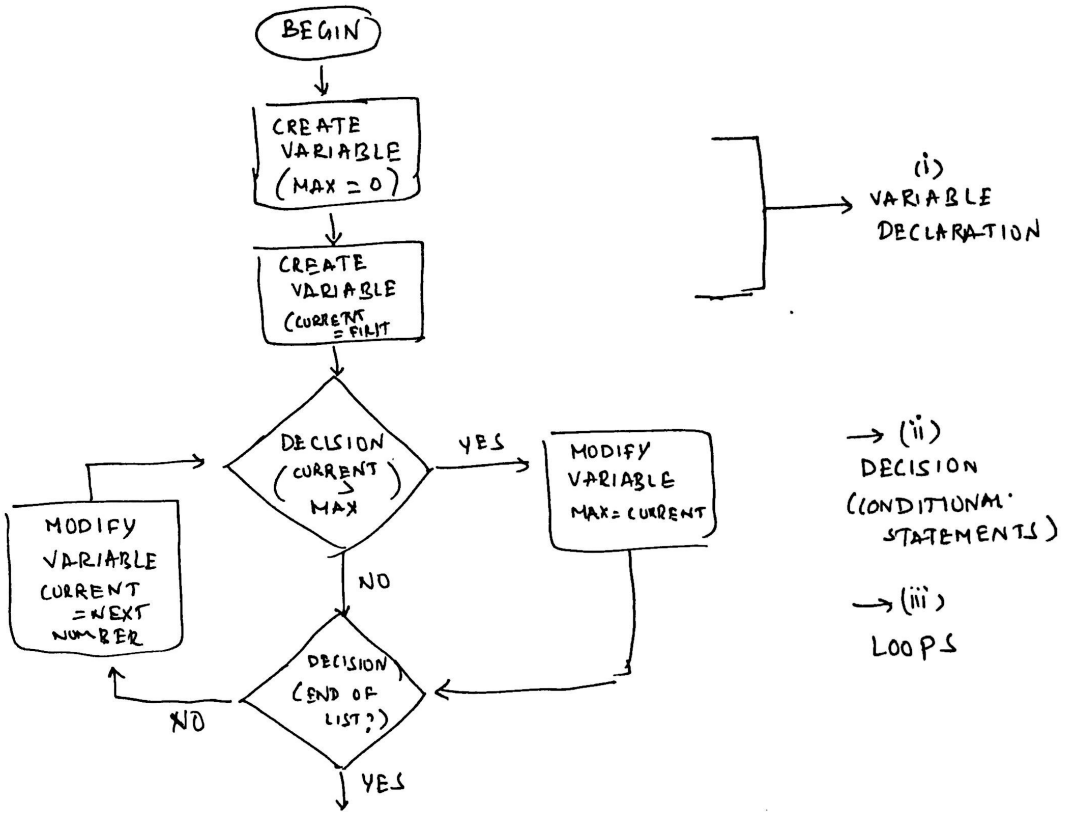

Lets begin!

- Declare the list of numbers in your programming language of choice.

  Numbers = [1 3 7 4 3 4 3 3 7 3]

- Create a max variable.
- Create current number and save the first value in the list in it.
- Print numbers, max and current_number.
- Comment out the print statements

# Hands on: Conditional statements



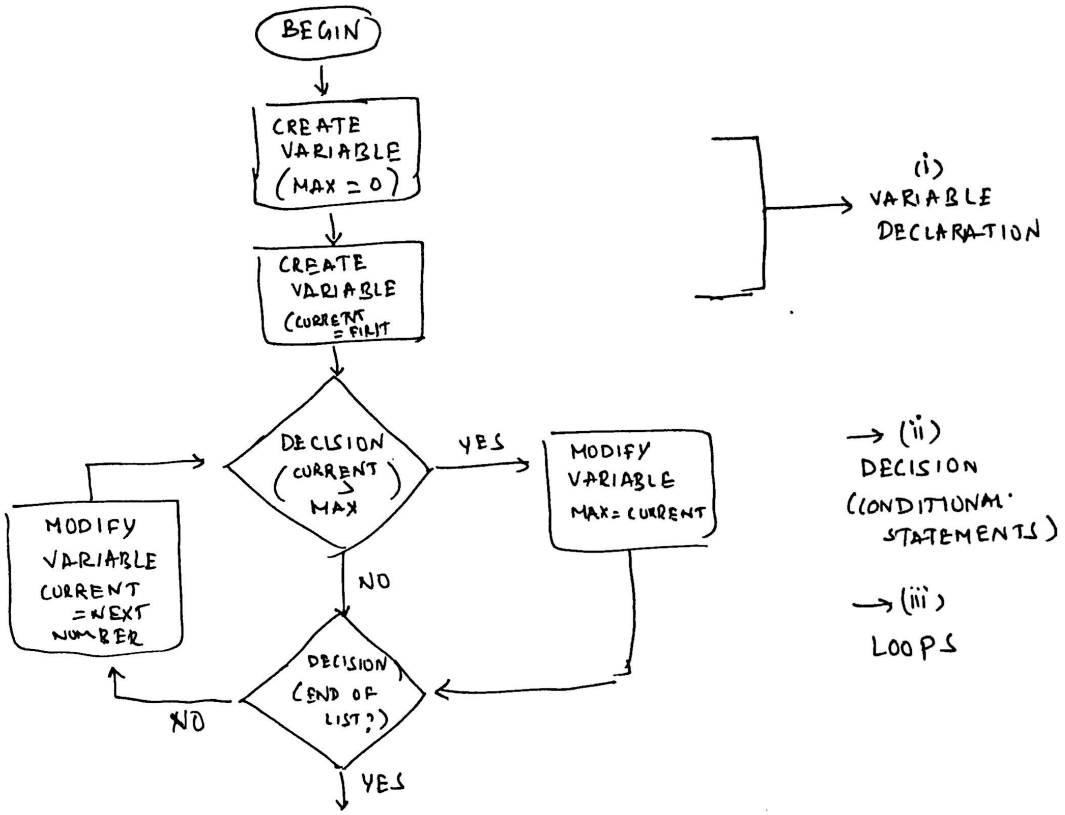

- Implicit in the algorithm is Boolean logic.
- Pseudo code:

```
if (a == 10):
        Do this.
else if (a == 20):
        Do this.
else:
        Do this
```

- Try this in your preferred language.

- Change the max from zero to the first number using a conditional statement.

# Hands on: Loops (for and while loops)




Loops allow us to perform the same computation multiple times.

- For loop (pseudo code)

  for (numbers in list):
  
  Keep doing this.

- While loop (pseudo code)

  while (current_number ~= last number):
  
  Keep doing this

- Complete coding up the algorithm and print out the maximum.

# Hands on session (coding time!)

- Max number algorithm - write a computer program to solve:
  - Here are some example sets to try and work on:
    - [49 46 63 62 25 14  8 62 35 36]
    - [841238 619645 209407 435618 832714 887185 164080 939827 179879 517613]
    - [-630 -260 -298 -538 -176 -596  -23 -693 -791 -271]
- Find the first 100(0) prime numbers.
- Find the first 100 numbers in the fibonacci sequence
  - You will learn a nice trick in programming if you finish this (:D).
- Find the factorial of an arbitrary number n.
  - Bonus points - Do this without using for/while loops (same trick as above).
- Advanced: Translate DNA sequence to protein sequence.
- Advanced: Sort the above numbers in ascending order.
- Challenge: Tower of hanoi
  - Figure out steps to move disks from one pole to the other, for any arbitrary number of disks n.
  - Bonus points - Do this without using for/while loops (same trick as above).

Fun set of videos (will help with the grammar): Crash course - computer science.
(approx 120 min total.)

# Questions or feedback?

Next session:

- File input output
- Handling (relatively) massive data
- Plotting and visualization