

SD400: Problem Solving

Lesson 1: Introduction to problem solving with computational thinking

Wholeness

- We are going to study problem solving with computational thinking (writing algorithms) because it is the essence of programming. We tell a computer how to do things, take input, and to create an output from it.
- A computer program is a tool that allows us to do less and accomplish more. Similarly TM is a tool that allows us to do less (gain restful alertness), and accomplish more (be more effective in action afterwards).

Lesson Objectives

- What is Computer Science
- Computational thinking
- What are algorithms (flowcharts)
- What are variables (model)
- Input / Output

Goals in this Course

- We aim to support and enhance your thinking abilities in this course
- Never feel bad for struggling
 - This is how your brain grows
 - Feel bad for not asking for clarification
 - Feel bad for not trying a different angle
 - Feel bad for giving up early

Why Study Computer Science?

- Computer Science is not the best name for our subject
- Computer science is the systematic study of algorithms, data, and the principles that govern the design and operation of computers and computational systems. It encompasses a wide range of topics, including programming, software development, hardware design, artificial intelligence, and the analysis of complex problems through computational techniques.
- Computer science plays a crucial role in shaping our digital world, driving technological innovation, and solving real-world challenges by leveraging the power of computers and information technology

Information Processes

- Business
- Biology
- Physics
- Agriculture
- Engineering
- ***Life all around us***

Maharishi Quote

“The range of computer science is the whole range of activity because any activity emerges continues, and finds its fulfillment on the basis of some computing process”

CS and Education

- You can memorize math formulas, and apply them again during the exam.
- What do you memorize in CS?
 - What does this mean for exams?

Memorization

- For a lot of students education has been centered around memorization, not reasoning
- The reality is you can look most things up online anyway
- Critical and Creative Thinking skills may be under-developed, often no attention or guidance has been given to them

Memorization

- Some memorization is always needed
 - You can't write a book if you don't know the alphabet
- But after the basics memorization is often laziness, why think and understand if you can memorize
 - Math formulas
 - Memorize the formula and you don't actually have to understand it, you don't actually have to be able know how it really works.

Critical & Creative Thinking

- Is memorizing good for critically analyzing something new, understanding how this new thing works?
- Is memorizing “how to do something” a good thing for creativity, for making something new?

World needs creative thinking!

- Everything is becoming smart
- Jobs are changing
- Everything is being automated
 - Robotic assistance (customer care)
 - Manufacturing
 - Financial transactions
 - Driverless vehicles
 - Advertising
 - etc...

Let's talk a little about 'Thinking'

- It requires a specific type of thinking.

Both:

- critical analysis (breaking into parts)
 - creative synthesis (merging solutions to parts into a whole)
-
- Solutions come from within:
 - Never memorized, but created from inside

CS and Thinking

- Analysis (breaking things down)
 - What is the problem?
 - What are its parts?
 - Can a part be broken into more parts?
- Synthesis (creating something new)
 - Can I solve a part?
 - How does the solution for this part relate to the solutions of other parts
 - How can I make a new whole (a solution)?

Main Point 1

- Studying Computer Science is all about gaining Analysis and Synthesis skills, central to critical and creative thinking.
- *Science of creative intelligence*: Critical thinking promotes curiosity, which encourages creativity to support innovation and productivity. This is essential since the nature of life is to grow.

What is an Algorithm?

- An algorithm is a sequence of instructions to achieve a task
 - Cooking Recipes are algorithms
 - Driving instructions are algorithms
- Write some instructions for how someone, who hasn't been in this building before could find the bathroom (starting from where you are sitting)

Self Referral

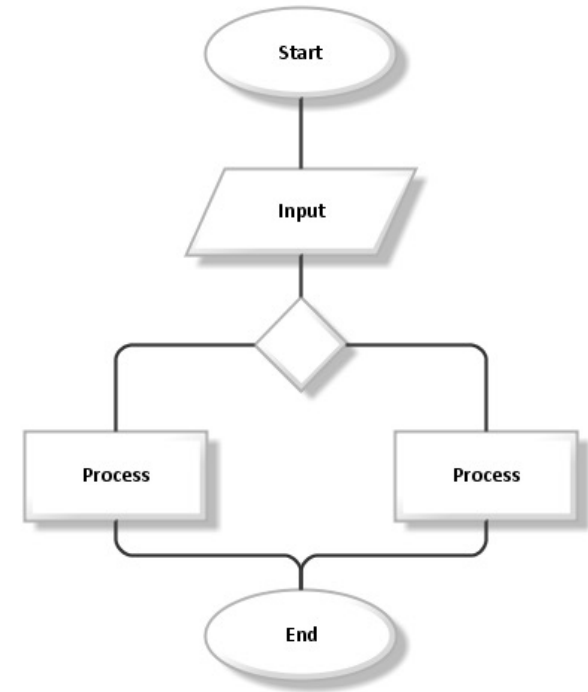
- Did you notice how this process was very self referral?
 - Writing the instructions mostly involved: How would I do that?
- Did you notice that there are different possibilities?
 - Leave the door open, or close it when you leave?
 - Describe the distance in feet, steps, or meters?
- These are some of the most important points:
 - There is never one solution / no “the solution”

Sequence






- Why is the sequence important?

Flowcharts

- A flowchart is a type of diagram that represents a workflow process.
- can also be defined as a representation of an algorithm, a step-by-step approach to solve a task.



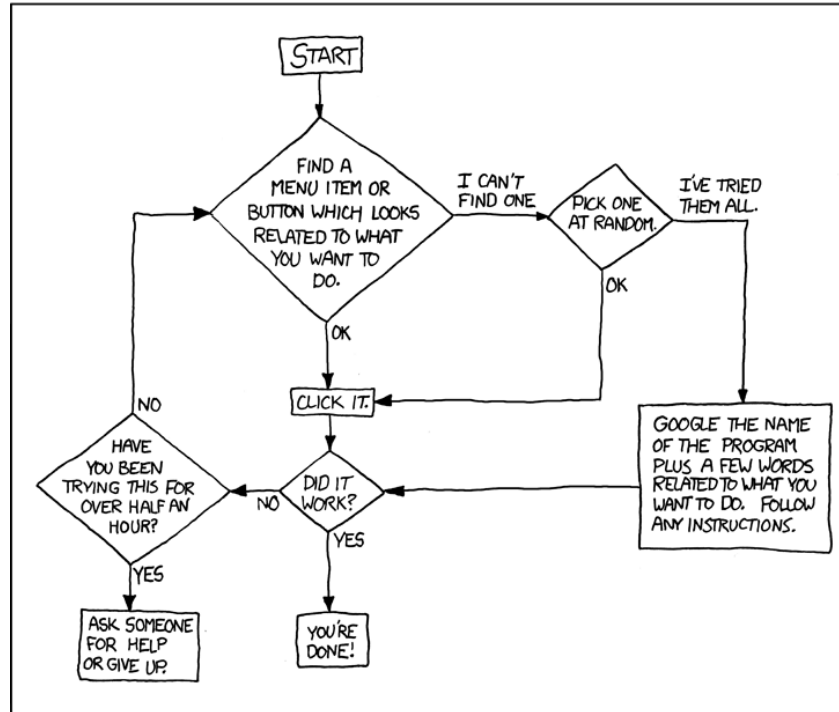
Flowchart symbols

Symbol	Name	Function
	Start/end	An oval represents a start or end point.
	Arrows	A line is a connector that shows relationships between the representative shapes.
	Input/Output	A parallelogram represents input or output.
	Process	A rectangle represents a process.
	Decision	A diamond indicates a decision.

Flowchart

DEAR VARIOUS PARENTS, GRANDPARENTS, CO-WORKERS,
AND OTHER "NOT COMPUTER PEOPLE."

WE DON'T MAGICALLY KNOW HOW TO DO EVERYTHING IN EVERY
PROGRAM. WHEN WE HELP YOU, WE'RE USUALLY JUST DOING THIS:



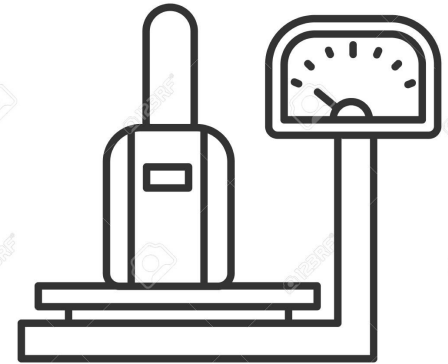
PLEASE PRINT THIS FLOWCHART OUT AND TAPE IT NEAR YOUR SCREEN.
CONGRATULATIONS; YOU'RE NOW THE LOCAL COMPUTER EXPERT!

Flow Chart

- Flowcharts are a tool that allows people to express what the steps for a process are
 - <http://xkcd.com/518/>
 - <https://xkcd.com/627/>
 - <http://xkcd.com/1195/>

Exercise: Make a Flowchart

- Luggage scale: if the weight is less or equal than 50 lbs., then no fee. If not, charge a fee.



Main Point 2

- An algorithm is a sequence of instructions. When people write an algorithm, they primarily think of 'how would I do that?'.
- *Science of creative intelligence*: Just like all of reality arises through the self interacting dynamics of the unified field, algorithms arise through the self interacting dynamics of our own consciousness

Telling the
computer to
left and right



Realizing the
prog. doesn't
understand
those
commands



Data

- Would the algorithm you wrote be able to work on a computer?
 - Does a computer know what left or right is?
 - Does a computer know what or where a door is?
 - Realizing the program doesn't understand left and right

Nothing

- The computer doesn't know anything
 - About anything at all
- The computer can store things
 - Data items that its told to store
 - Algorithms (steps) aka programs

Model

- You have to create a model
 - Store data 'about' the world
 - An abstraction (just containing the things you need)
- Then have the computer manipulate it
 - Perhaps we have person on a map, and that person has X and Y coordinates
 - The person 'moves' by changing his X and Y

Model of the Building and of Our Person

- Lets make a drawing on this slide of a grid (x and y axis)
 - Draw the building layout, add my location, add the bathroom location
- Suggested solution
 - $X = 10$
 - $Y = 10$
 - $P = N$
 - Steps = ?
 - Current Position = (X, Y)
 - Restroom = (X, Y)

How do you Create a Model?

- What data do I need at the beginning?
- What data do I need at the end?
- What data will I manipulate to change the beginning to the end?
- (and what data will I need while manipulating?)
- It's always different, but always related to the task at hand. In our case:
 - X and Y of the person
 - X and Y of each segment of the walls

Variables

- X and Y are what we call variables
 - The computer manipulates (changes) them
 - To achieve the desired result

The Equals Sign

- Many programming languages use the equals sign for assignment (storage) into variables
 - This sometimes really confuses people up when starting CS
 - Doesn't make sense from a math background
- Important! The equal sign in most programming languages does not indicate equality
 - It indicates assignment

Assignment

- An assignment statement in CS is a two step process
 1. Turn the expression on the right hand side into a single value
 2. Store this value into the variable on the left hand side
- Any variables used on the right hand side will be turned into the value they represent (the value they hold)

Example

$X = 5$

$Y = -15$

$Z = X + Y$

$Y = Z$

print X

print Y

print Z

Exercise

What is the output of the following program

```
A = 10
```

```
B = 20
```

```
A = B
```

```
print A
```

```
print B
```

“Fix” the algorithm

- Lets think about the bathroom algorithm again
 - What would be our model
 - We would have to manipulate the X and Y of the person
 - Assuming a person cannot teleport
 - Assuming a person cannot walk through walls
 - At what X / Y would a person start?
 - At what X / Y would the bathroom be?
 - How big is each step?

Useful

- Is our bathroom program useful so far?
 - X and Y of the person are only inside the computer
 - How does this tell a person where to go?

Input and Output

- In order for a program to be useful for humans, it needs:
 - Input: receive something (from a human)
 - Output: give something (to a human)

Hello World

- The traditional 'first program' for any language
 - Output “Hello World”
 - As shown earlier in the lecture
- Please note that output text needs to be in quotes
 - We actually output an expression
 - Expressions can contain other instructions as well!
 - How the computer knows what is what:
 - Instructions are words outside of quotes
 - Inside quotes is text (data)

Input and Output

- Next we'll update our program to take input
 - Our name
- And then output
 - Hello [name]
- We will do this by using:
 - The concatenation operator + to combine strings (text data)
 - The variable where the name is stored in our expression

Exercise: Expand the Program

- Create a flowchart that asks for a person's name and age
 - First output "Hello " + name
 - And then output "Your age is: " + age
 - Notice both outputs go on the same line

New Lines

- What if we wanted to output 2 lines (1 for name, 1 for age)
- The computer has Special Control characters
 - These do not show up as text
 - Influence how the text is shown
- The most common of these is the newline
 - This is written as `\n`
 - The `\` before it indicates that this is not a normal `n`
 - Because `\` indicates the start of a control character, the only way to output an actual `\` is by typing `\\`

Manipulate input

Create new output

- The purpose of basically every computer program is to take some input and turn it into the desired output
- The problem we as programmers solve is figuring out what the instructions and data model need to be.
 - In order to change the given input to the desired output

Main Point 3

- Input and Output are how the computer interacts with humans.
- *Science of creative intelligence*: In general every action has an equal and opposite reaction. Our goal is to create the optimal reaction based on the users action.

Summary

- The world needs more people who think with reason and logic.
- Computer Science is all about thinking reason and logic by analyzing problems and synthesizing solutions.
- In this lecture we've seen:
 - What are algorithms
 - What are variables (model)
 - Input / Output