An **algorithm** is a procedure to solve a problem:

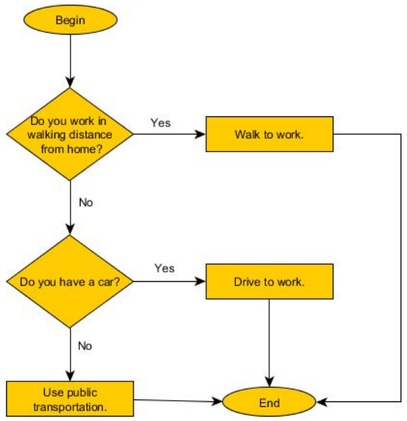
A series of steps that when followed solve a specific problem

A problem can be seen to have two components:
- Input
- Desired output

Inputs have sizes

The size of the input can affect the amount of steps (time) needed to solve a problem

This relationship between the size of the input and the time it takes to solve the problem is key focus of this course



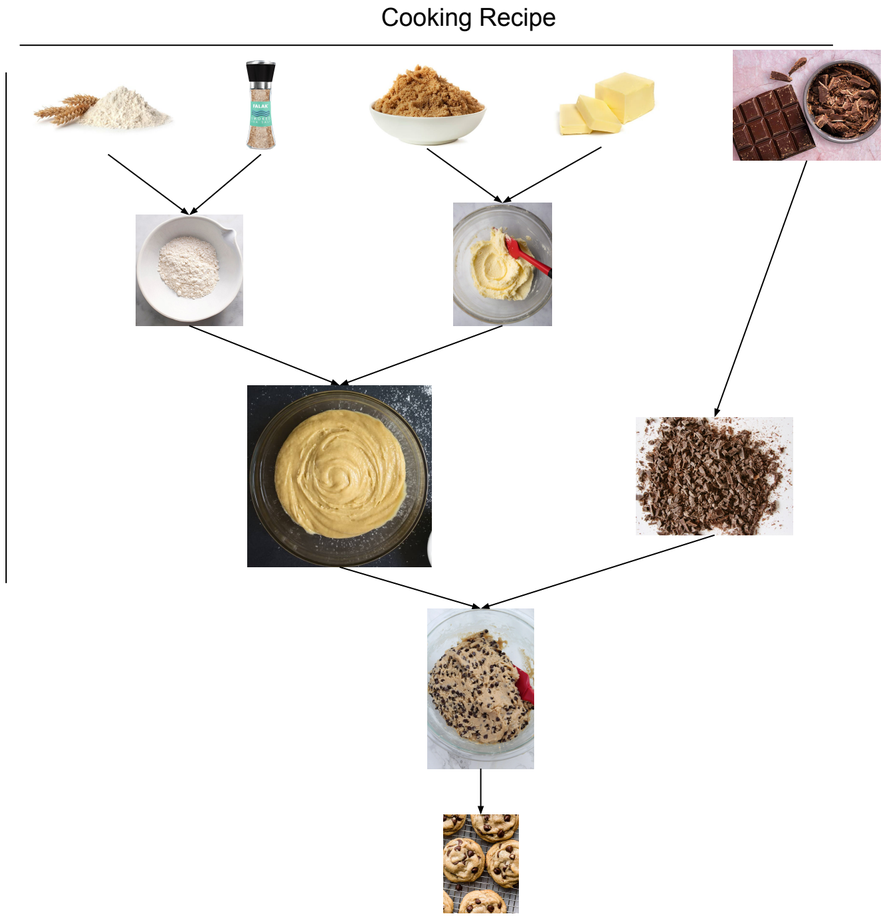
N: 5
Runtimes will be the same

log(n) * n

1024
512
256
128
64
32
16
8
4
2
1

10

Log (base 2) of 1024 = 10



Problem to Solve	What is N	How fast is the **best case** scenario to solve this problem	" " " **average case** " " "	" " " **worst case** " " "
Find the ace of spades	52	1	26	52
Count the cards in a deck	52	52 -> N	52 -> N	52
Check if the deck is in sorted order (4-A, 4-1, ..)	52	1	-	52
Check if a box of cards has cards inside	1	1	1	1
Solve the matching problem (Brute force)		N *N		
Solve the matching problem (Perfect memory)				
Following a cooking recipe (of a specific style)				

The matching problem