This is CS50


FROM HOT

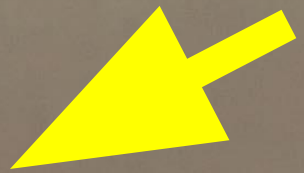
most CS50 students
have never taken CS before

what ultimately matters in this course is not so much where you end up relative to your classmates but where you end up relative to yourself when you began

```
/*
 * hello.c
 *
 * Assignment: Assignment 1
 *
 * Name: David Malan
 *
 * A program to print "Hello, CS50!" on the screen.
 */

#include    <stdio.h>

/*
 * main
 */

void        main ()
{

      printf ("Hello, CS50!\n");

      exit (0);
}

/*
 * end of hello.c
 */
```

(-2) for hello.out, we wanted output of hello, not of make.

```
/*
 * hello.c
 *
 * Assignment: Assignment 1
 *
 * Name: David Malan
 *
 * A program to print "Hello, CS50!" on the screen.
 */

#include    <stdio.h>

/*
 * main
 */

void        main ()
{

      printf ("Hello, CS50!\n");

      exit (0);
}

/*
 * end of hello.c
 */
```



(-2) for hello.out, we wanted output of hello, not of make.

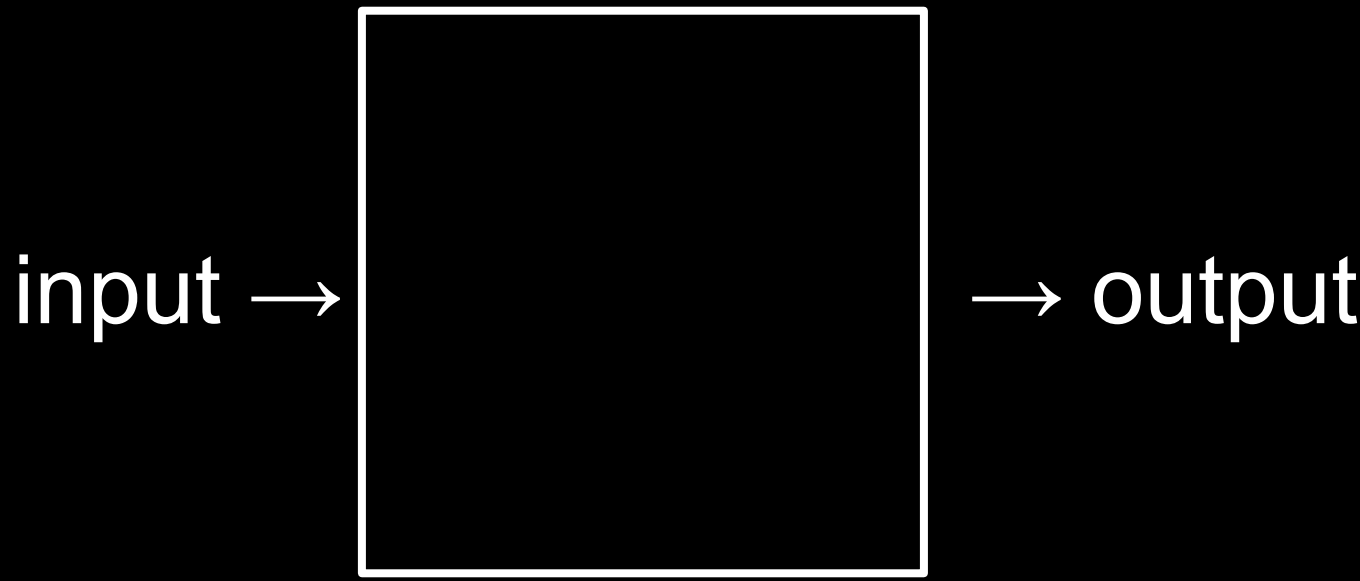

I took CS50.


SEC
WELCOME ABOARD
206

computer science

computational thinking

problem solving


input →
→ output

unary

base-1

base-2

binary

binary digit

bi t

bit

1

base-10

decimal

123

123

10 1

123

100 10 1

123

100 10 1

123

100 × 1

100 10 1

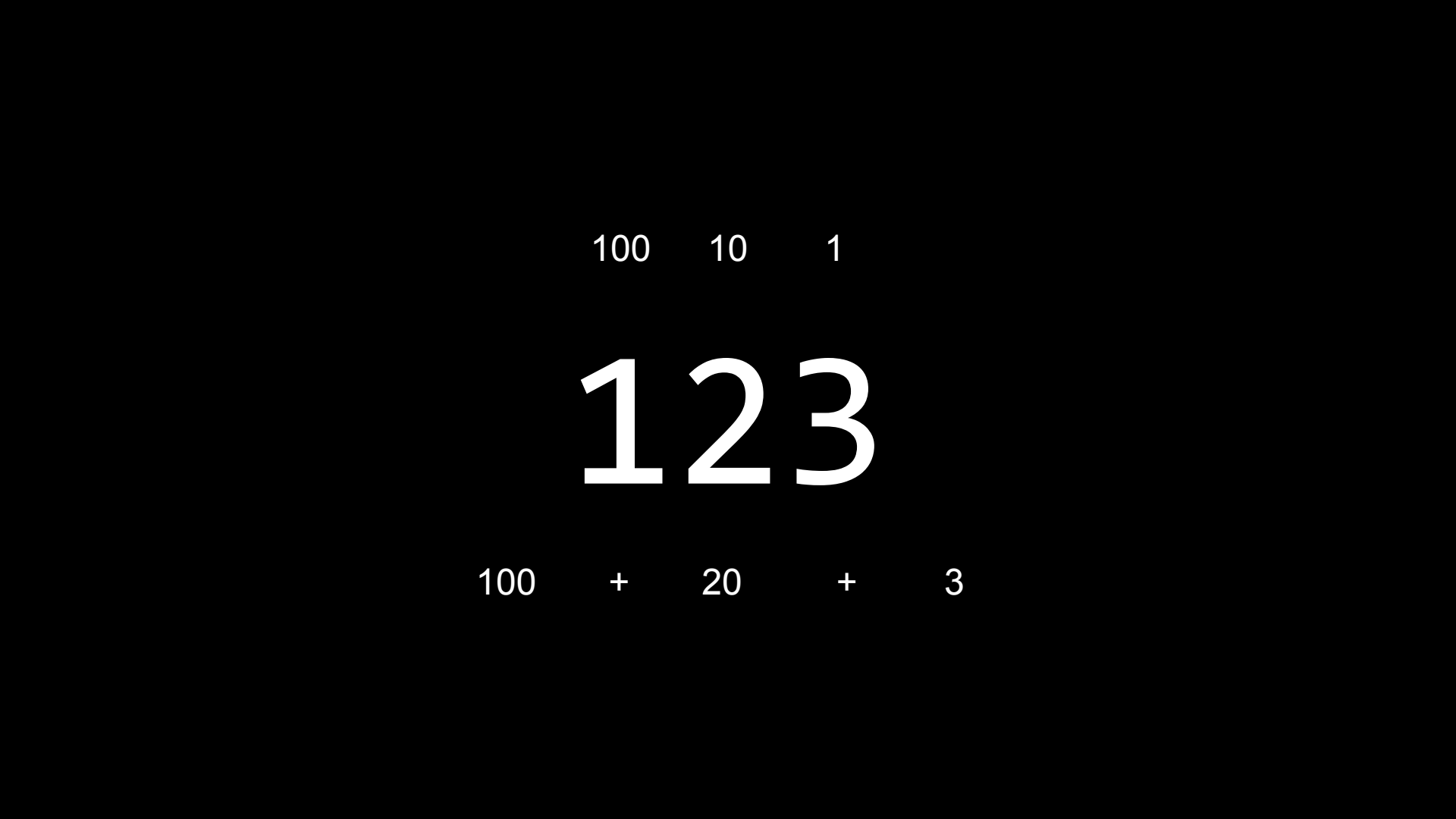
123

100 × 1 + 10 × 2

100 10 1

123

100 × 1 + 10 × 2 + 1 × 3


100 10 1
123
100 + 20 + 3

123

100 10 1

###

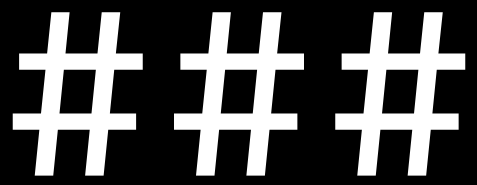
10^2 10^1 10^0

###

2^2 2^1 2^0

###

4 2 1




4 2 1
000

4 2 1

001

4 2 1

010

4 2 1

011

4 2 1

100

4 2 1

101

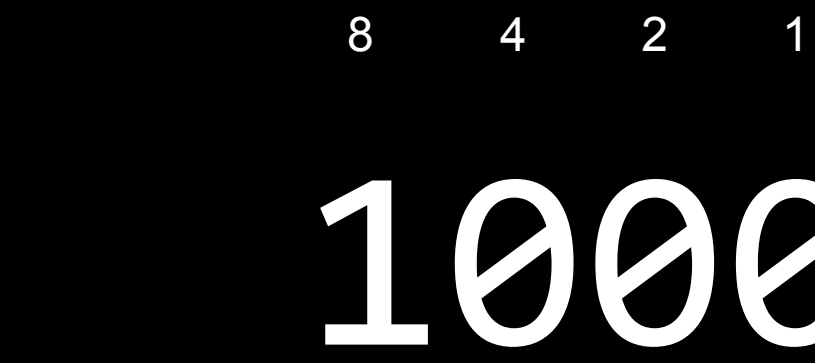
4 2 1

110

4 2 1

111

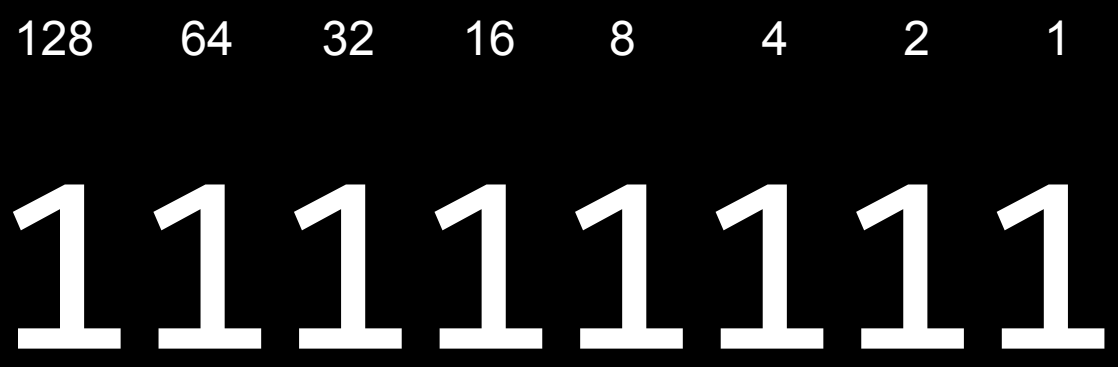

4 2 1
000


8 4 2 1
1000

byte

128 64 32 16 8 4 2 1

0 0 0 0 0 0 0 0


128 64 32 16 8 4 2 1
11111111

A

128 64 32 16 8 4 2 1

01000001

65

ASCII

0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	`	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	BEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(	56	8	72	H	88	X	104	h	120	x
9	HT	25	EM	41	)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VT	27	ESC	43	+	59	;	75	K	91	[	107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	|
13	CR	29	GS	45	-	61	=	77	M	93	]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL

0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	`	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	BEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(	56	8	72	H	88	X	104	h	120	x
9	HT	25	EM	41	)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VT	27	ESC	43	+	59	;	75	K	91	[	107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	|
13	CR	29	GS	45	-	61	=	77	M	93	]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL

01001000 01001001 00100001

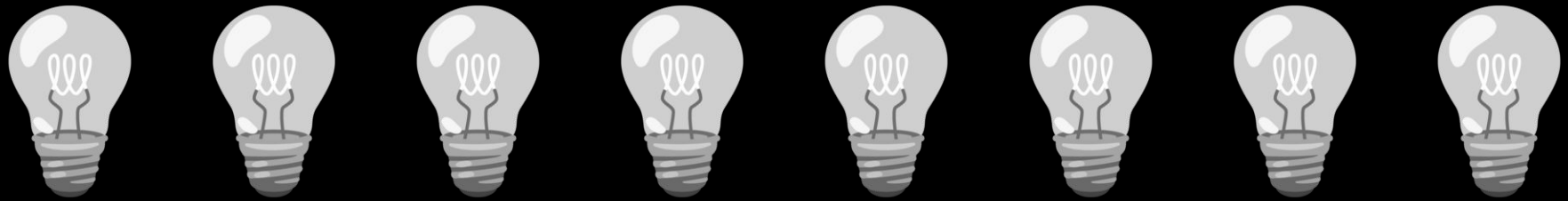
72 73 33

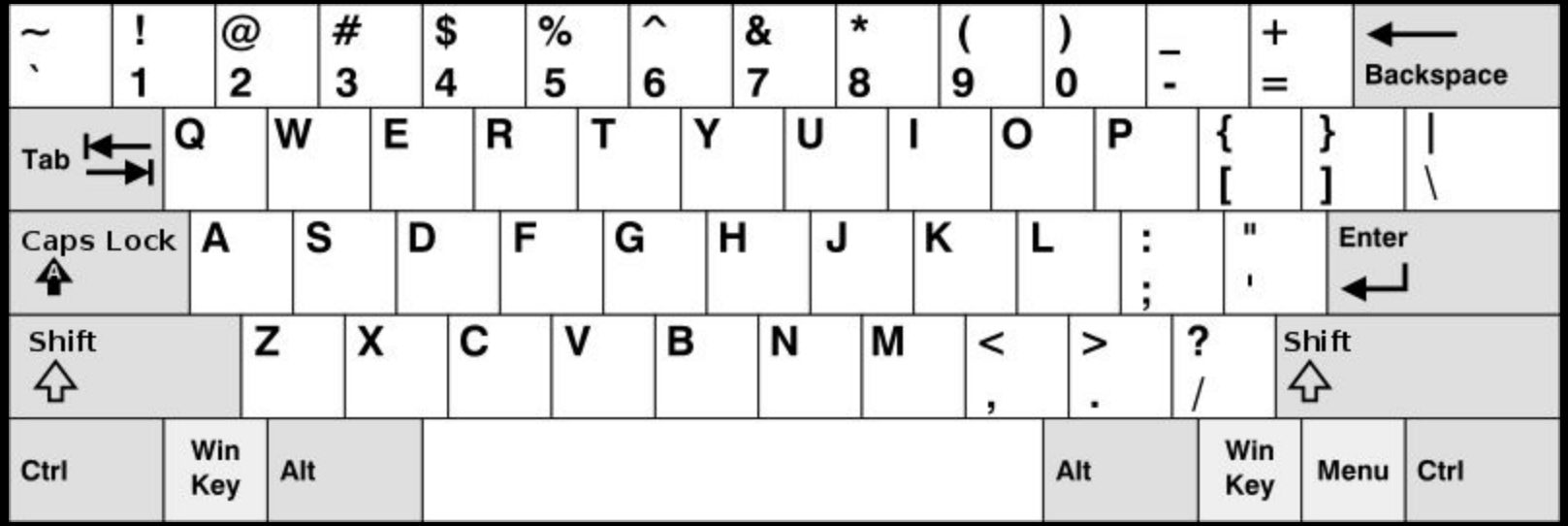
0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	`	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	BEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(	56	8	72	H	88	X	104	h	120	x
9	HT	25	EM	41	)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VT	27	ESC	43	+	59	;	75	K	91	[	107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	|
13	CR	29	GS	45	-	61	=	77	M	93	]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL

0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	`	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	BEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(	56	8	72	H	88	X	104	h	120	x
9	HT	25	EM	41	)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VT	27	ESC	43	+	59	;	75	K	91	[	107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	|
13	CR	29	GS	45	-	61	=	77	M	93	]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL

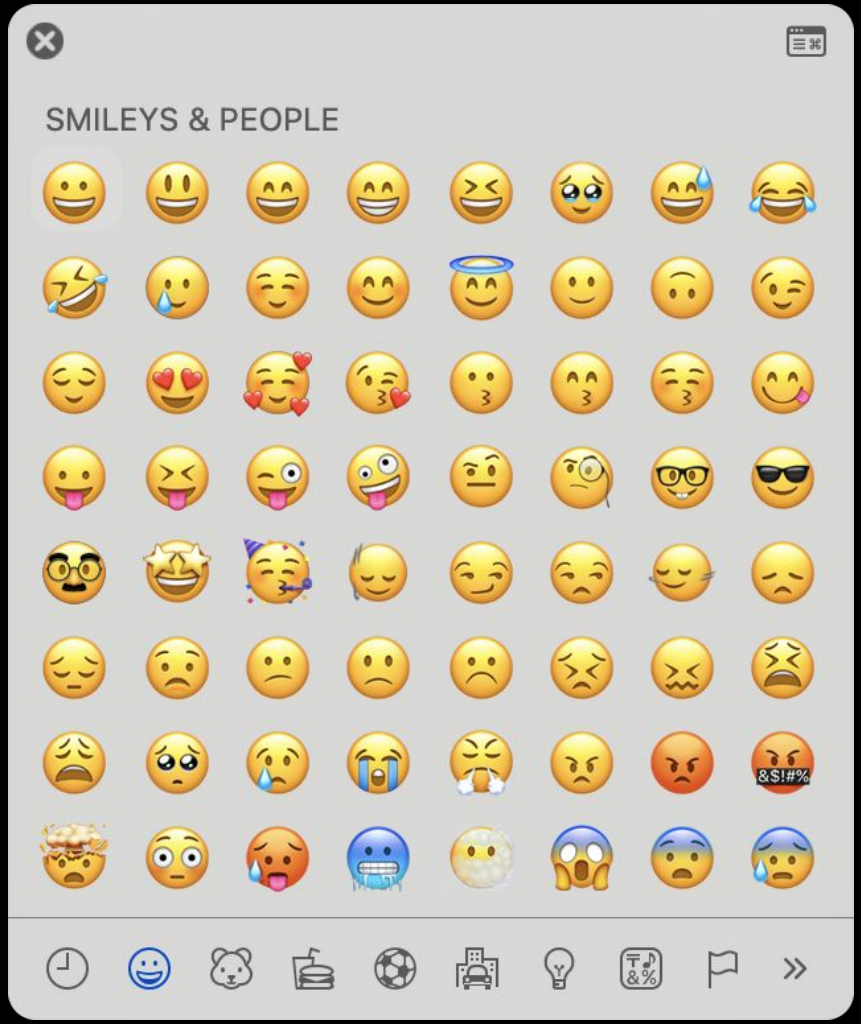
H	I	!
72	73	33

0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	`	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	BEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(	56	8	72	H	88	X	104	h	120	x
9	HT	25	EM	41	)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VT	27	ESC	43	+	59	;	75	K	91	[	107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	|
13	CR	29	GS	45	-	61	=	77	M	93	]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL




~ `
! 1
@ 2
# 3
$ 4
% 5
^ 6
& 7
* 8
(9
) 0
_ -
+ =
Backspace
Tab
Q
W
E
R
T
Y
U
I
O
P
{ [
}]
| \
Caps Lock
A
S
D
F
G
H
J
K
L
: ;
" '
Enter
Shift
Z
X
C
V
B
N
M
< ,
> .
? /
Shift
Ctrl
Win Key
Alt
Alt
Win Key
Menu
Ctrl

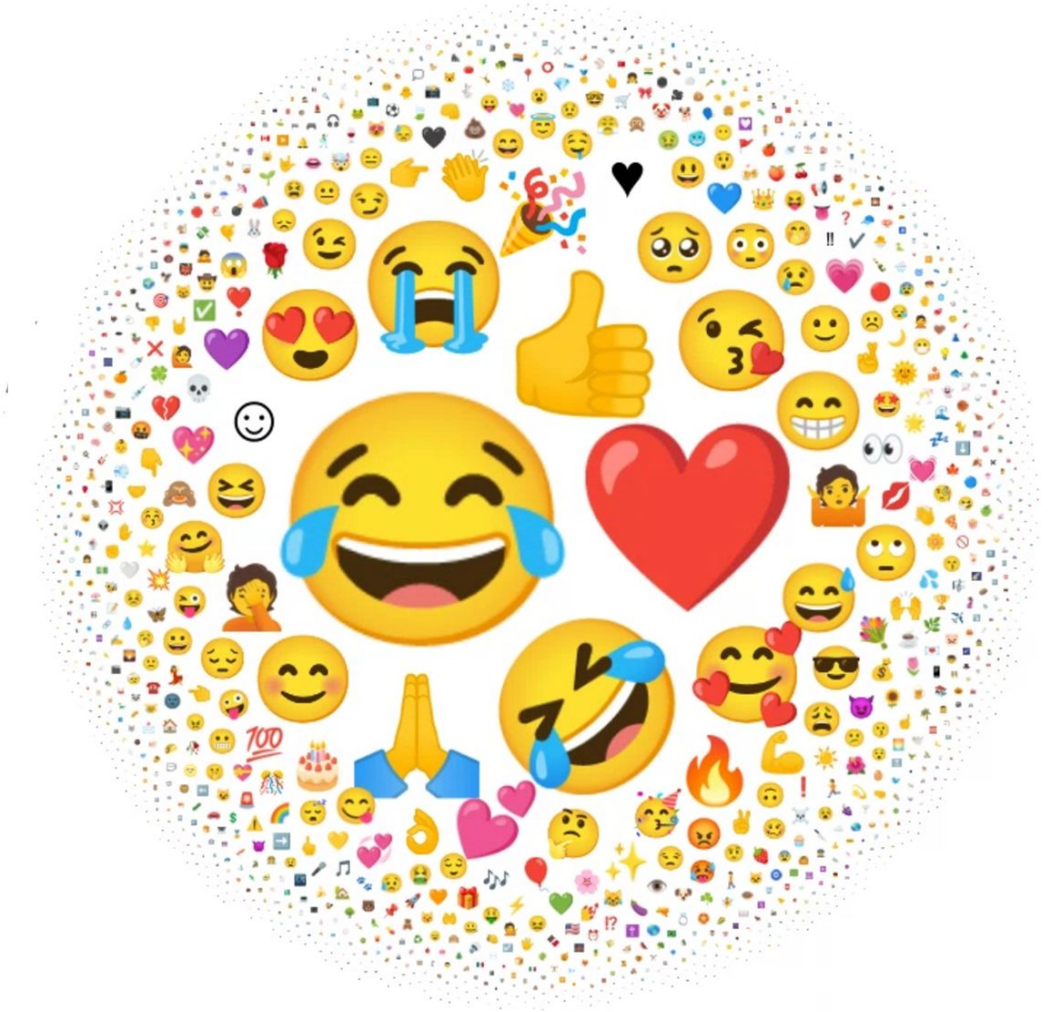

à á â ä æ ã å ā
1 2 3 4 5 6 7 8
a


SMILEYS & PEOPLE

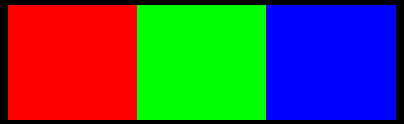
Unicode

11110000100111111001100010000010

4036991106

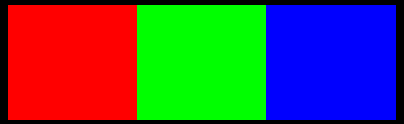


RGB



72 73 33


72
73
33



face-with-tears-of-joy_1f602.png

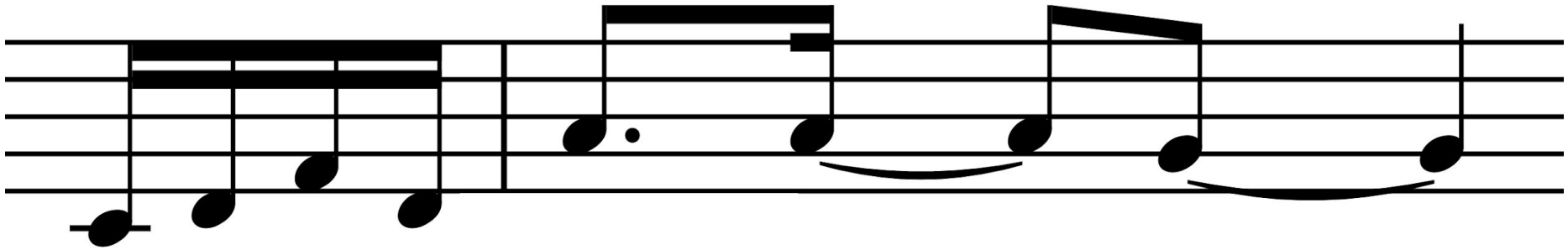
Search

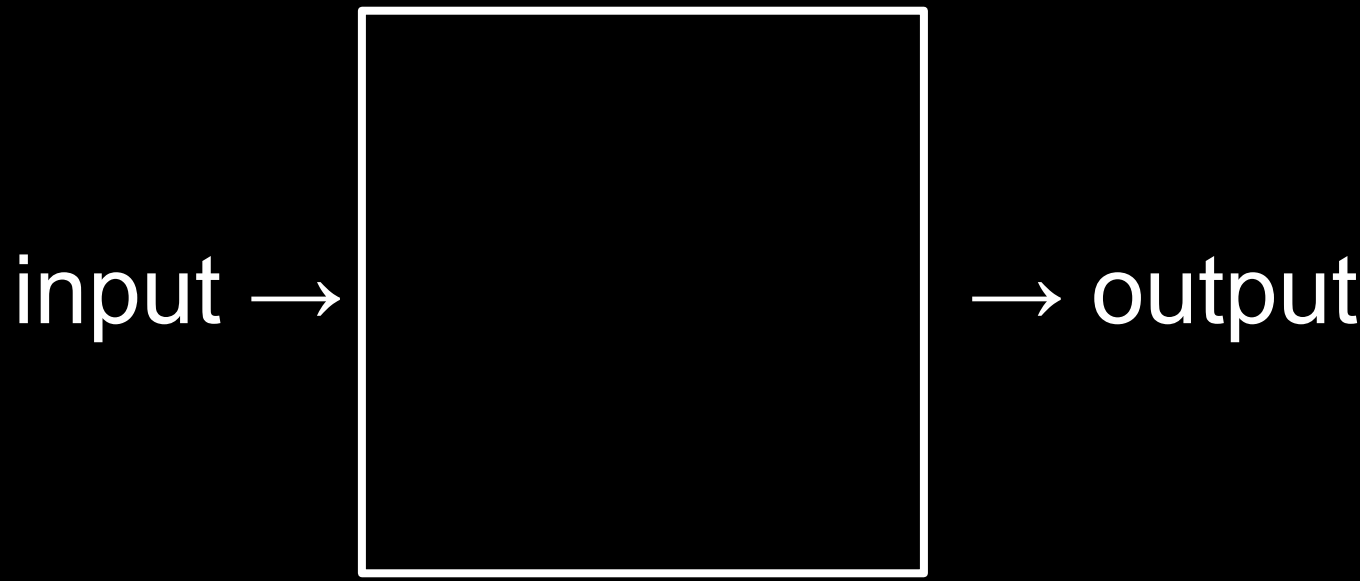
face-with-tears-of-joy_1f602.png

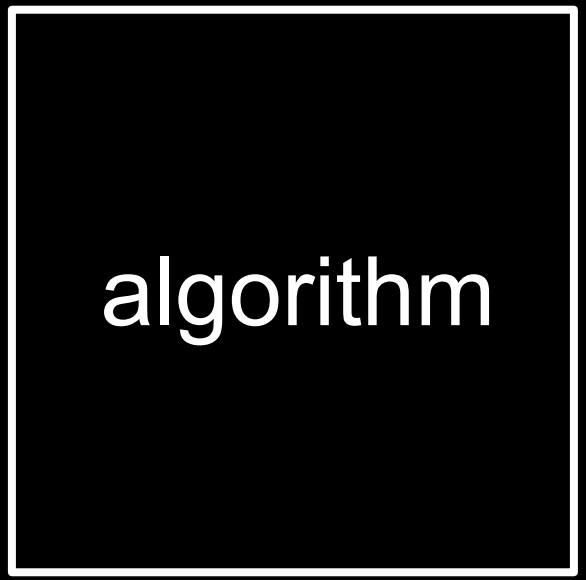
Search

face-with-tears-of-joy_1f602.png

Search



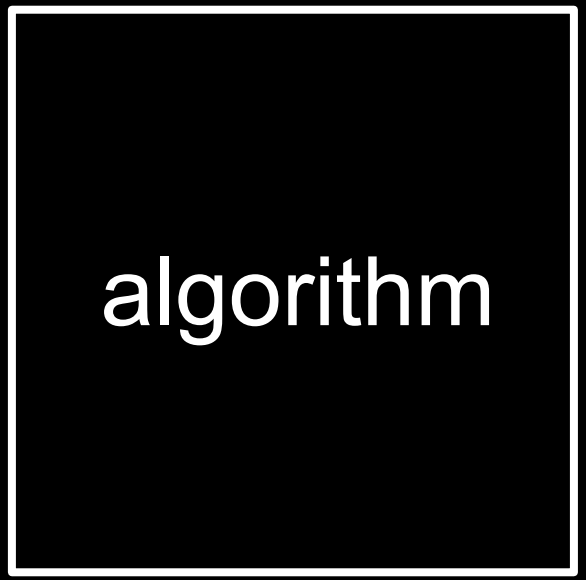

input →
→ output

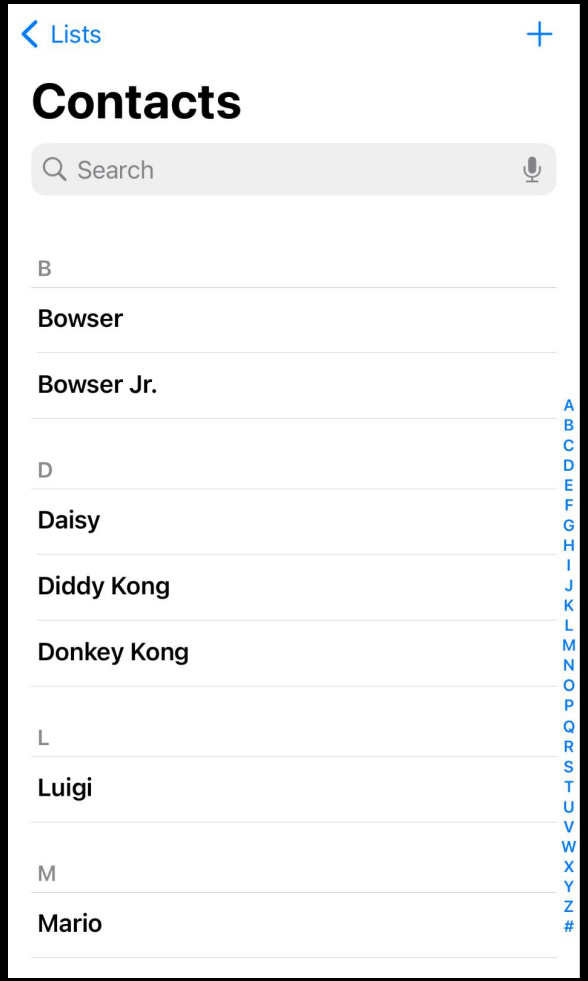

algorithm

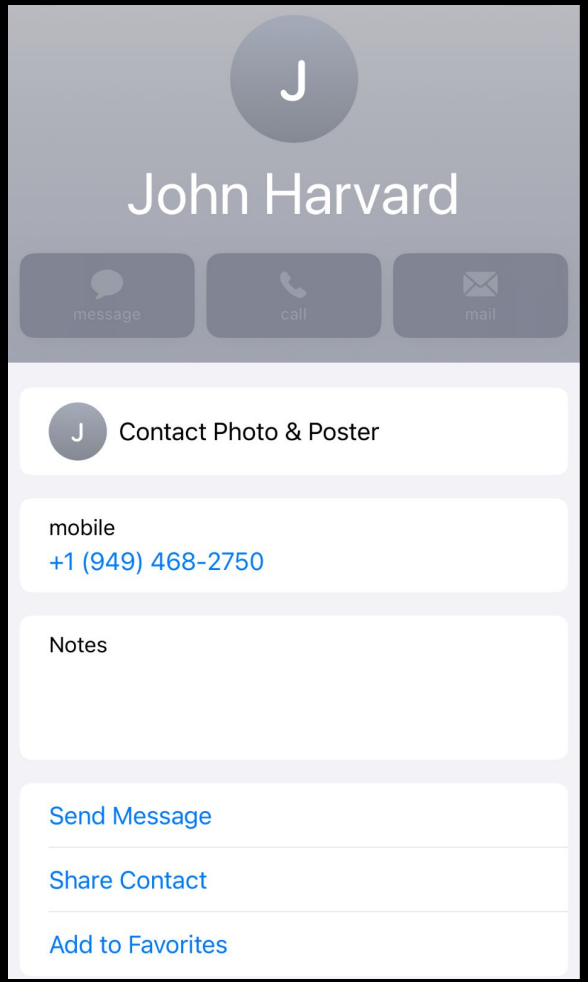
- planning a trip
- hierarchy is how we describe, organize, and build big things.

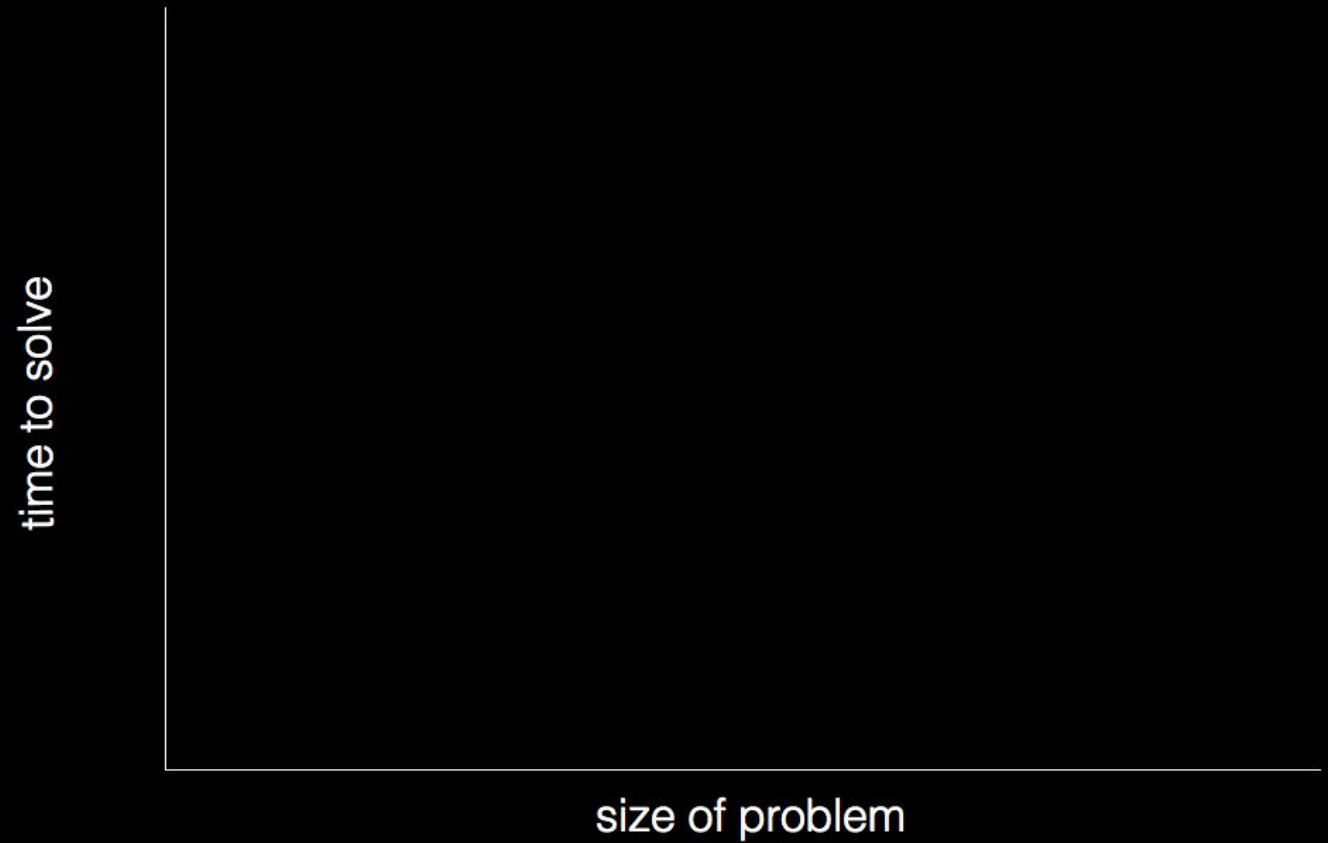
Algorithm: a precise description of how to do something

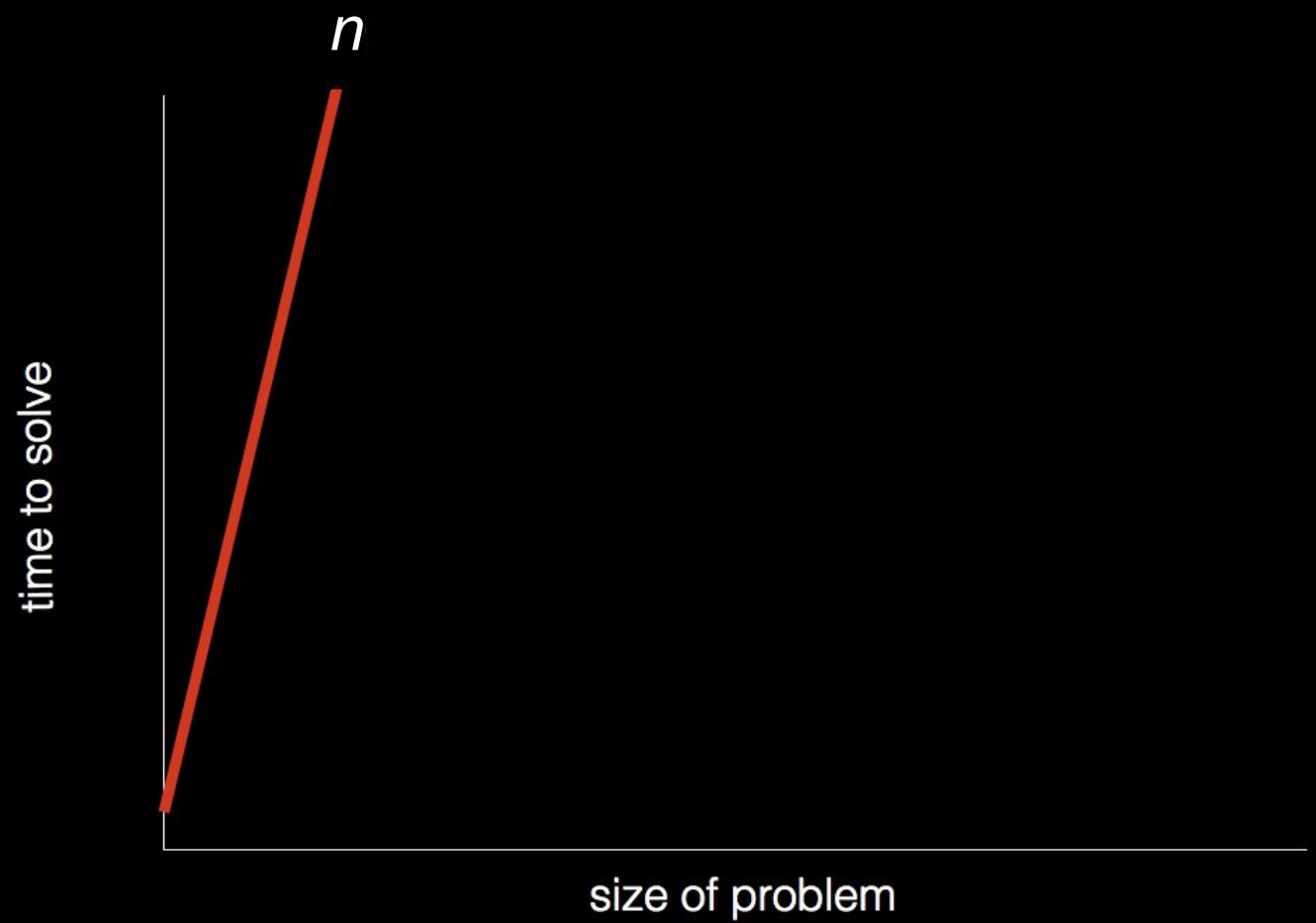
- **precise** sequence of operations (using a specified set of basic "primitive" operations)
- **correct**!
- **finite**: they will stop eventually

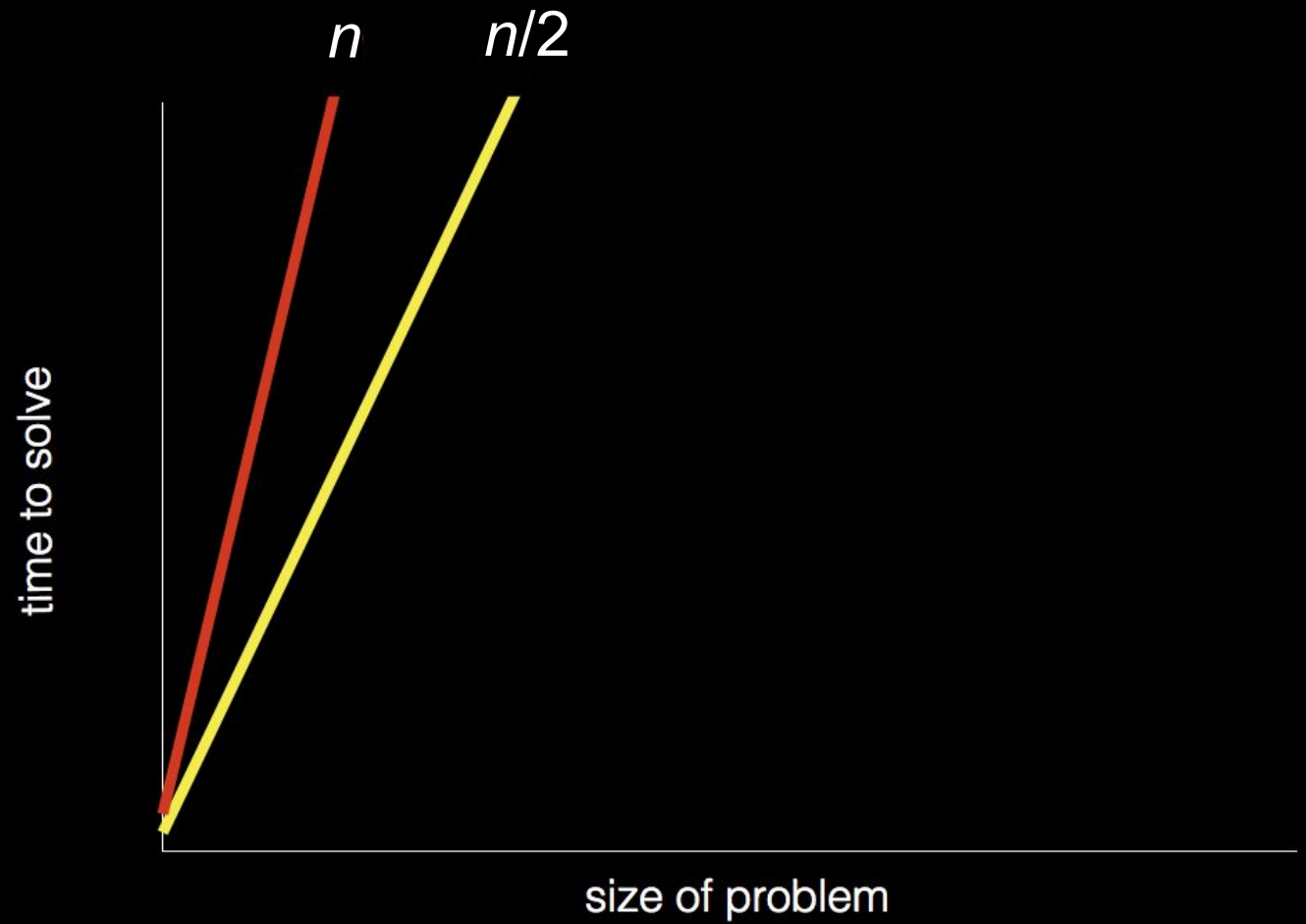

algorithm

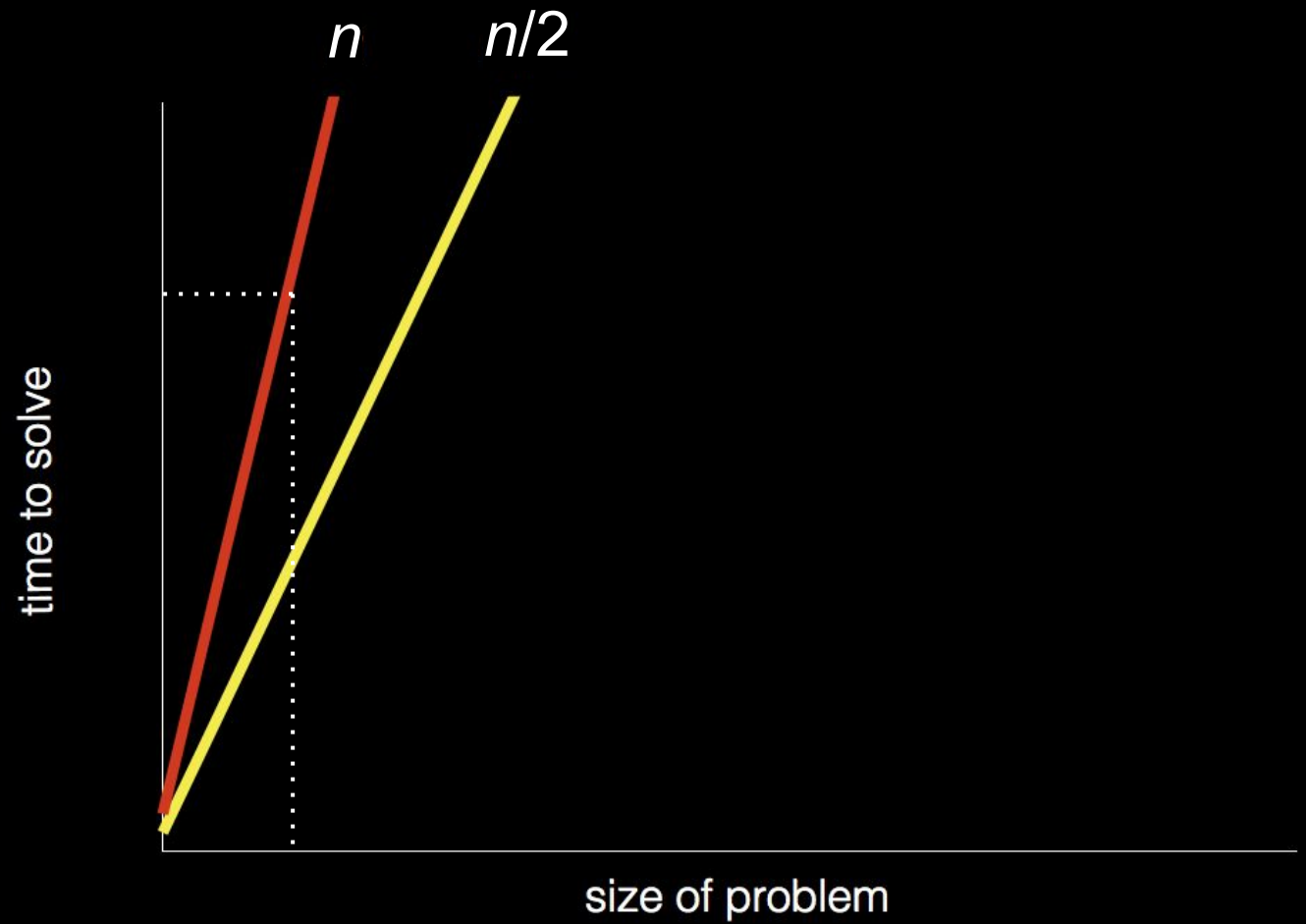

Lists
Contacts
Search
B
Bowser
Bowser Jr.
D
Daisy
Diddy Kong
Donkey Kong
L
Luigi
M
Mario
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
#

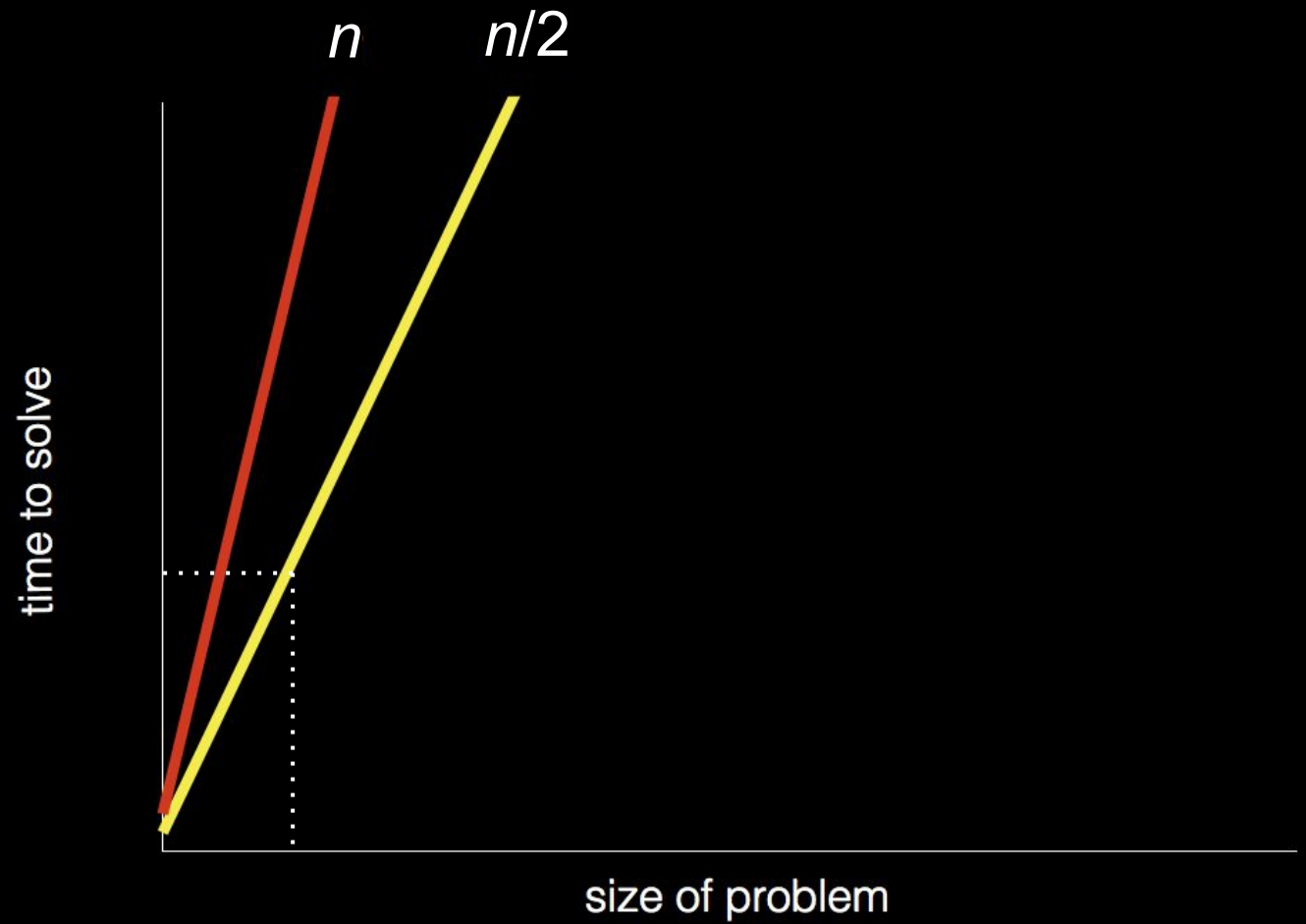

J
John Harvard
message
call
mail
J
Contact Photo & Poster
mobile
+1 (949) 468-2750
Notes
Send Message
Share Contact
Add to Favorites

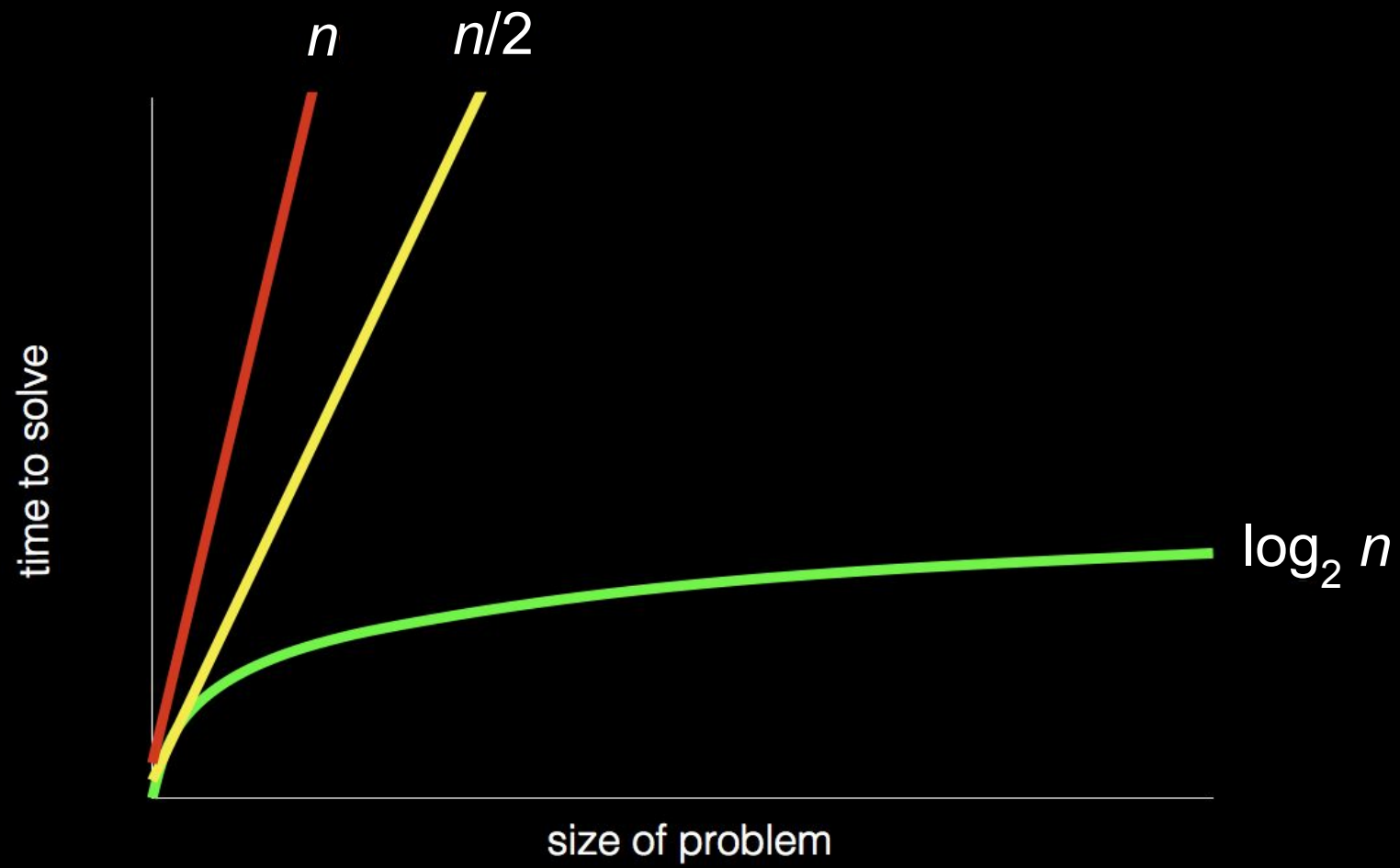

time to solve
size of problem

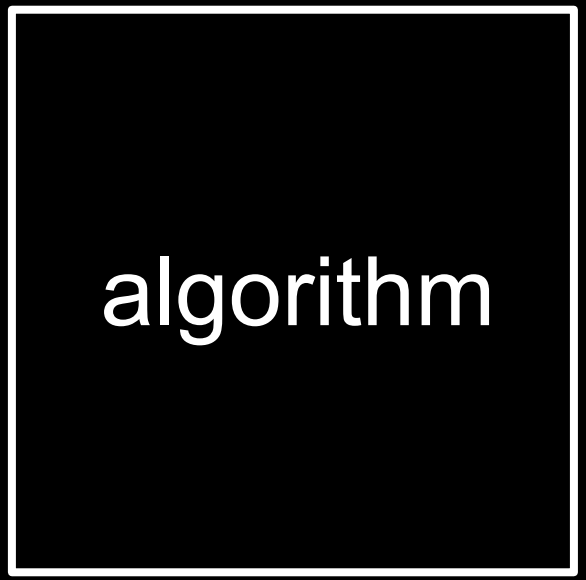

n
time to solve
size of problem

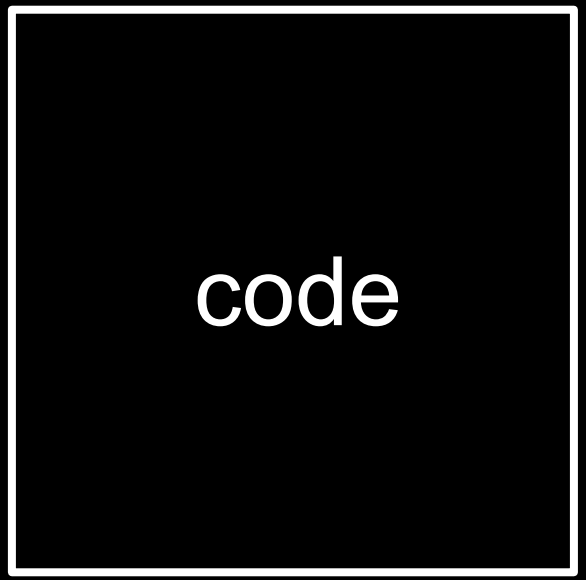

n
n/2
time to solve
size of problem

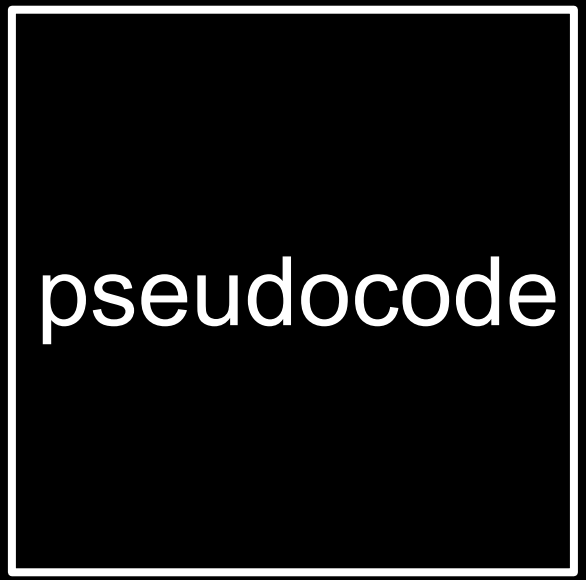

n
n/2
time to solve
size of problem


n
n/2
time to solve
size of problem


n
n/2
time to solve
log2 n
size of problem


algorithm


code


pseudocode

```
1   Pick up phone book
2   Open to middle of phone book
3   Look at page
4   If person is on page
5       Call person
6   Else if person is earlier in book
7       Open to middle of left half of book
8       Go back to line 3
9   Else if person is later in book
10      Open to middle of right half of book
11      Go back to line 3
12  Else
13      Quit
```

```
1   Pick up phone book
2   Open to middle of phone book
3   Look at page
4   If person is on page
5       Call person
6   Else if person is earlier in book
7       Open to middle of left half of book
8       Go back to line 3
9   Else if person is later in book
10      Open to middle of right half of book
11      Go back to line 3
12  Else
13      Quit
```

```
1   Pick up phone book
2   Open to middle of phone book
3   Look at page
4   If person is on page
5       Call person
6   Else if person is earlier in book
7       Open to middle of left half of book
8       Go back to line 3
9   Else if person is later in book
10      Open to middle of right half of book
11      Go back to line 3
12  Else
13      Quit
```

```
1   Pick up phone book
2   Open to middle of phone book
3   Look at page
4   If person is on page
5       Call person
6   Else if person is earlier in book
7       Open to middle of left half of book
8       Go back to line 3
9   Else if person is later in book
10      Open to middle of right half of book
11      Go back to line 3
12  Else
13      Quit
```

```
1   Pick up phone book
2   Open to middle of phone book
3   Look at page
4   If person is on page
5       Call person
6   Else if person is earlier in book
7       Open to middle of left half of book
8       Go back to line 3
9   Else if person is later in book
10      Open to middle of right half of book
11      Go back to line 3
12  Else
13      Quit
```

- functions
  - arguments, return values, variables
- conditionals
- Boolean expressions
- loops
- ...

artificial intelligence

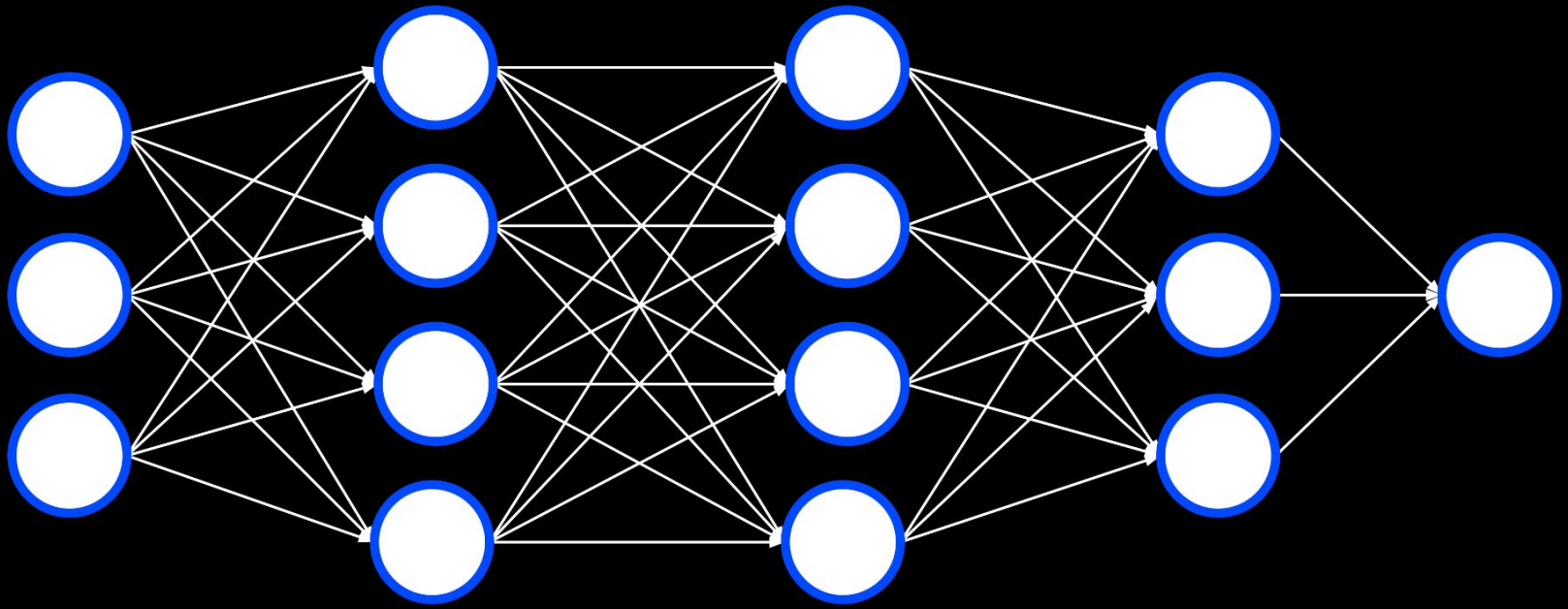
```
If student says hello
    Say hello
```

```
If student says hello
    Say hello
Else if student says goodbye
    Say goodbye
```

```
If student says hello
    Say hello
Else if student says goodbye
    Say goodbye
Else if student asks how you are
    Say well
```

```
If student says hello
    Say hello
Else if student says goodbye
    Say goodbye
Else if student asks how you are
    Say well
Else if student asks why 111 in binary is 7 is decimal
    …
```

large language models




CS50

Not Reasonable

Using AI-based software other than CS50's own…

Reasonable

Using CS50's own AI-based software…

CS50 Duck

cs50.ai

Visual Studio Code for CS50

cs50.dev

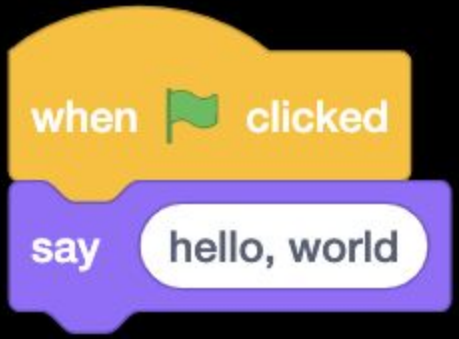
```c
#include <stdio.h>

int main(void)
{
    printf("hello, world\n");
}
```


01111111010001010100110001000110000000100000000100000001000000000000000000000000000000000000000000
00000000000000000000000000000001000000000111110000000000000010000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000000000000000000100000000000000001010000000000000010000000
00010101010100100010001001111001010100100010000111110110000010000001100011100000010001001110001110100
10001011111000000000000000000000000000000000000000000000000000000000000000000000000000010110000000000001110100000
00000000000000000000000000001001000101111110000000000000000000000000000000000000000000000000000000000
0000000001001000...01111111010001010100110001000110000000100000000100000001000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000011000000000011111000000000000000100000000000000000
00000001100000000001111000000000000000000000000000000000000000000000000000000000000000010000000000000000000000000
00000000000000000000000000000000000010100000110010000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000010000000000000011100000000000000011100000000010000000000000000011100000
00000000110010000000000000010000000000000000000000000000000010100000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000...00101111011011000110100101100010011000110010111001110011011011110
00101110001101100010000000101111011101010111001101110010001011110110110001101001011000100010111011110
00001110000011011001011111001101100011010000101101011011000110100101101110011101010111100001011010110
01110110111001110101001011110110110001101001011000100110001101011111011011100110111101101110011100110101
10100001100001011100100110010101100100001011100110000100100000010000001000001010100110101111101001110
01000101010001010100010001000101010001000010000000101000001000000010111101101100011010010110001000101111
11011110000011100000110110010111110011011000110100001011010110110001101001011011100111010101111000001011
01010110011101101110011101010010111101101100011001000010110101101100011010010110111001110101011110000010
10110101111000001110000011011000101101001101100011010...

break

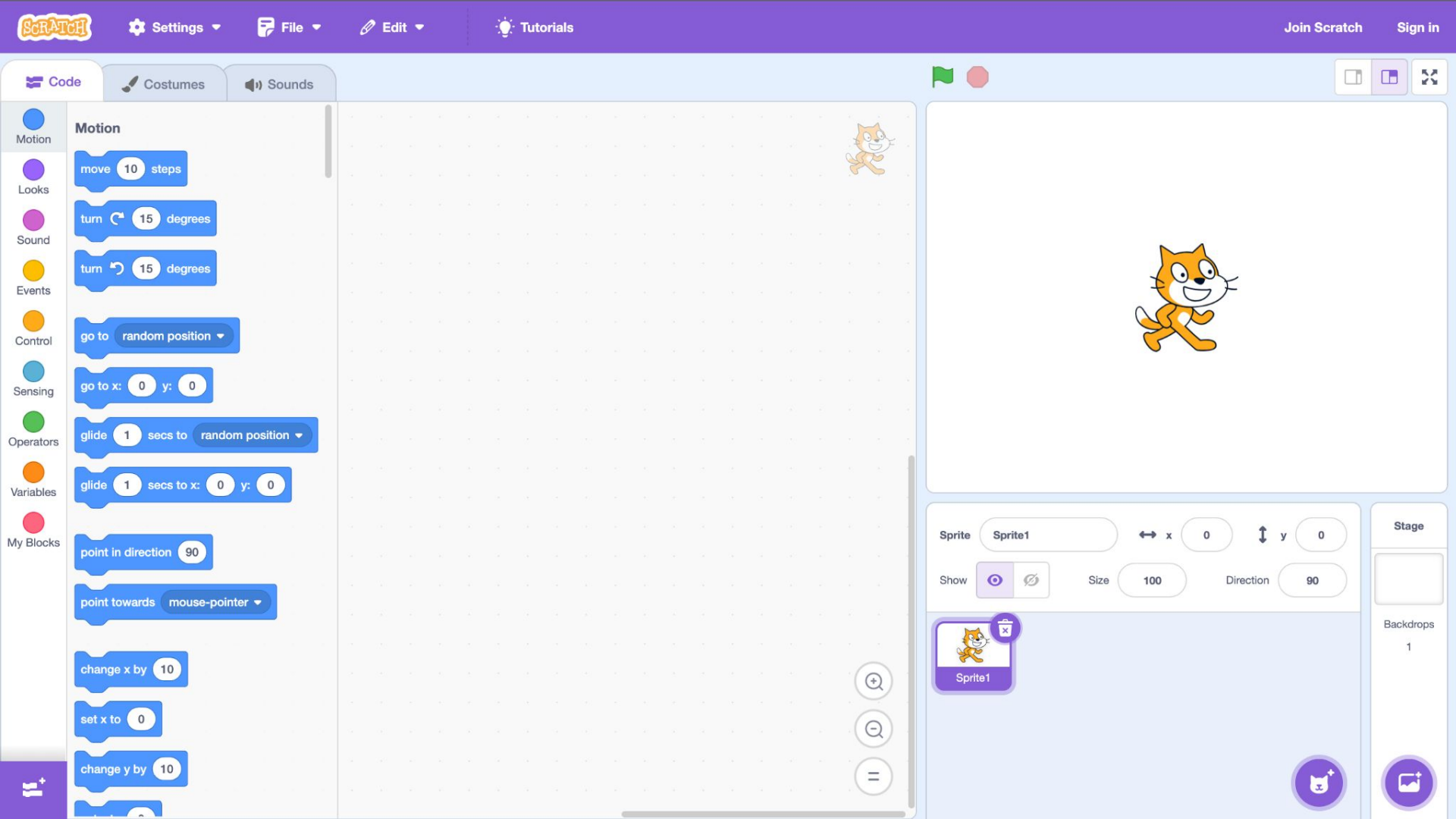

01111111010001010100110001000110000000100000000100000001000000000000000000000000000000000000000000
00000000000000000000000000000001000000000111110000000000000010000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000000000000000000100000000000000001010000000000000010000000
00010101010100100010001001111001010100100010000111110110000010000001100011100000010001001110001110100
10001011111000000000000000000000000000000000000000000000000000000000000000000000000000010110000000000001110100000
00000000000000000000000000001001000101111110000000000000000000000000000000000000000000000000000000000
0000000001001000...01111111010001010100110001000110000000100000000100000001000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000011000000000011111000000000000000100000000000000000
00000001100000000001111000000000000000000000000000000000000000000000000000000000000000010000000000000000000000000
00000000000000000000000000000000000010100000110010000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000010000000000000011100000000000000011100000000010000000000000000011100000
00000000110010000000000000010000000000000000000000000000000010100000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000...00101111011011000110100101100010011000110010111001110011011011110
00101110001101100010000000101111011101010111001101110010001011110110110001101001011000100010111011110
00001110000011011001011111001101100011010000101101011011000110100101101110011101010111100001011010110
01110110111001110101001011110110110001101001011000100110001101011111011011100110111101101110011100110101
10100001100001011100100110010101100100001011100110000100100000010000001000001010100110101111101001110
01000101010001010100010001000101010001000010000000101000001000000010111101101100011010010110001000101111
11011110000011100000110110010111110011011000110100001011010110110001101001011011100111010101111000001011
01010110011101101110011101010010111101101100011001000010110101101100011010010110111001110101011110000010
10110101111000001110000011011000101101001101100011010...

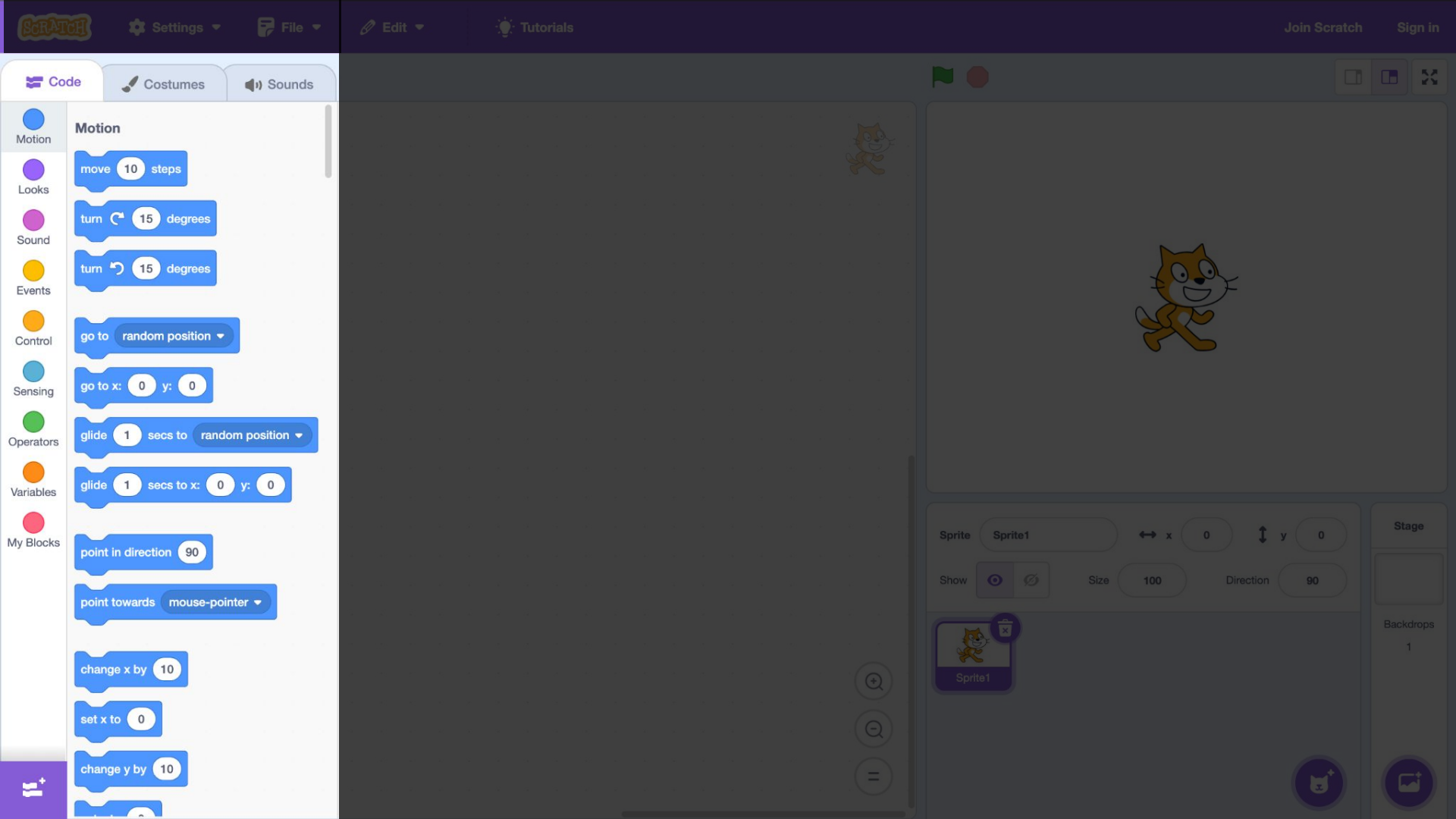
```c
#include <stdio.h>

int main(void)
{
    printf("hello, world\n");
}
```

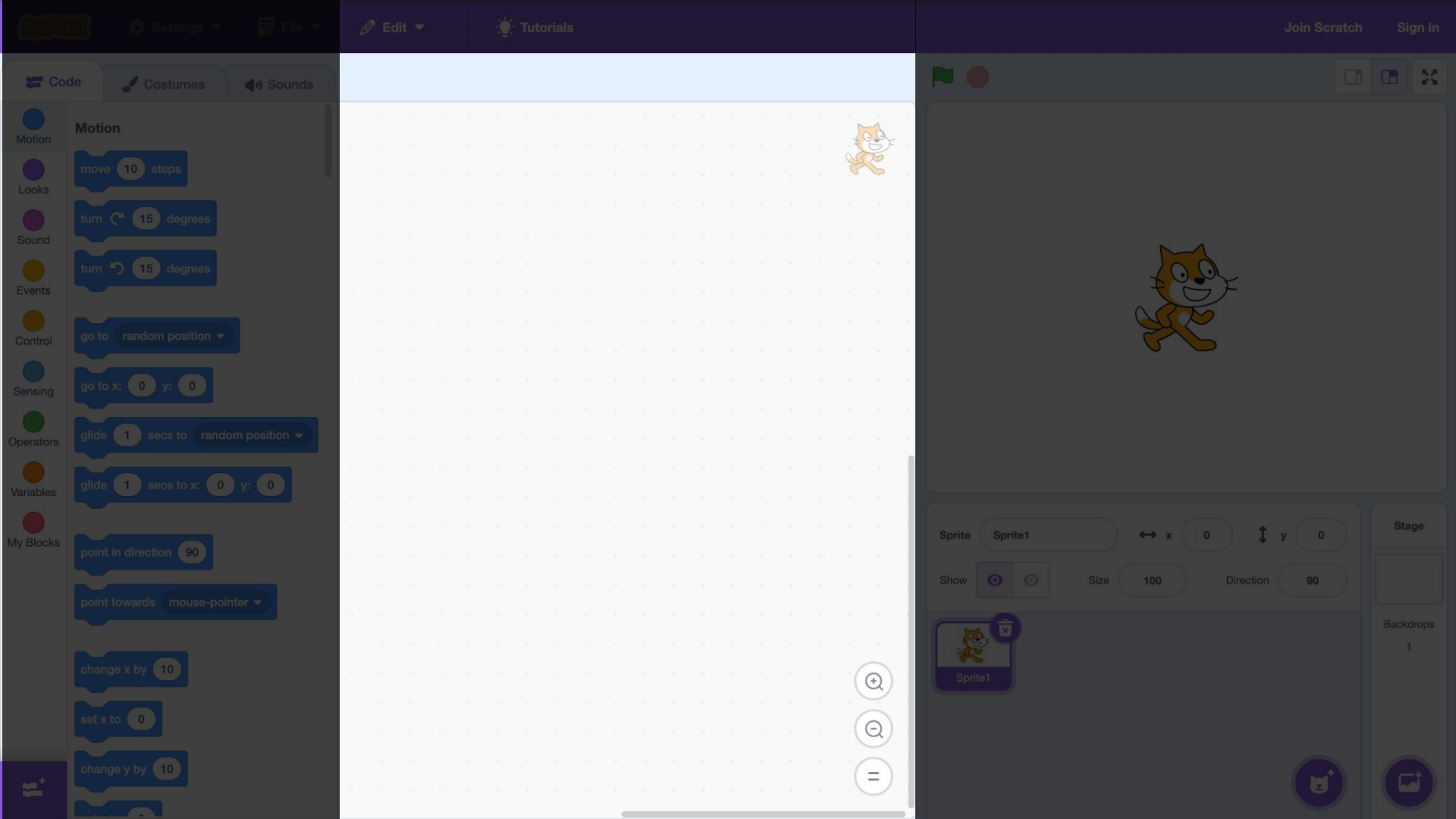

when clicked
say
hello, world

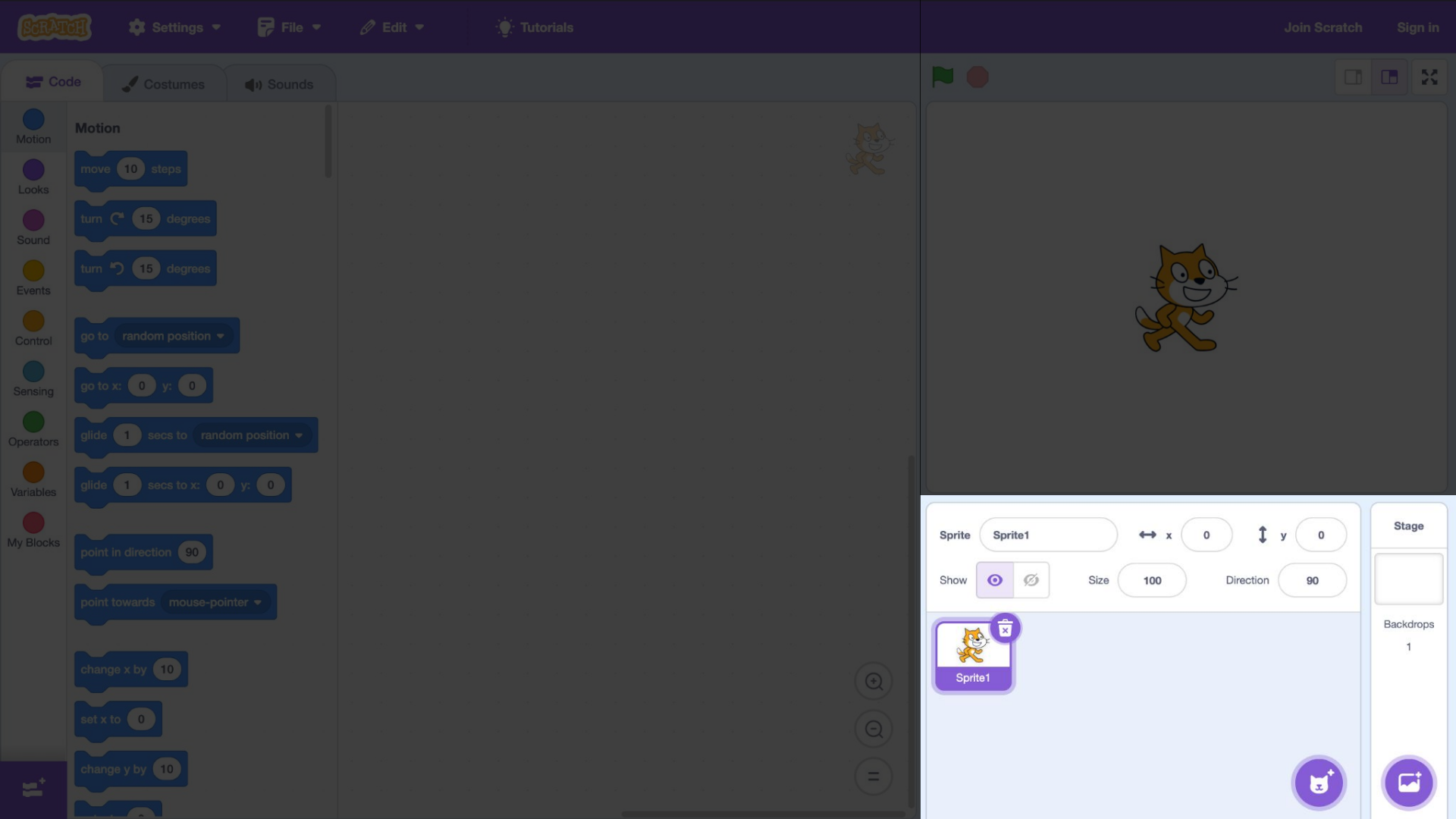
Scratch

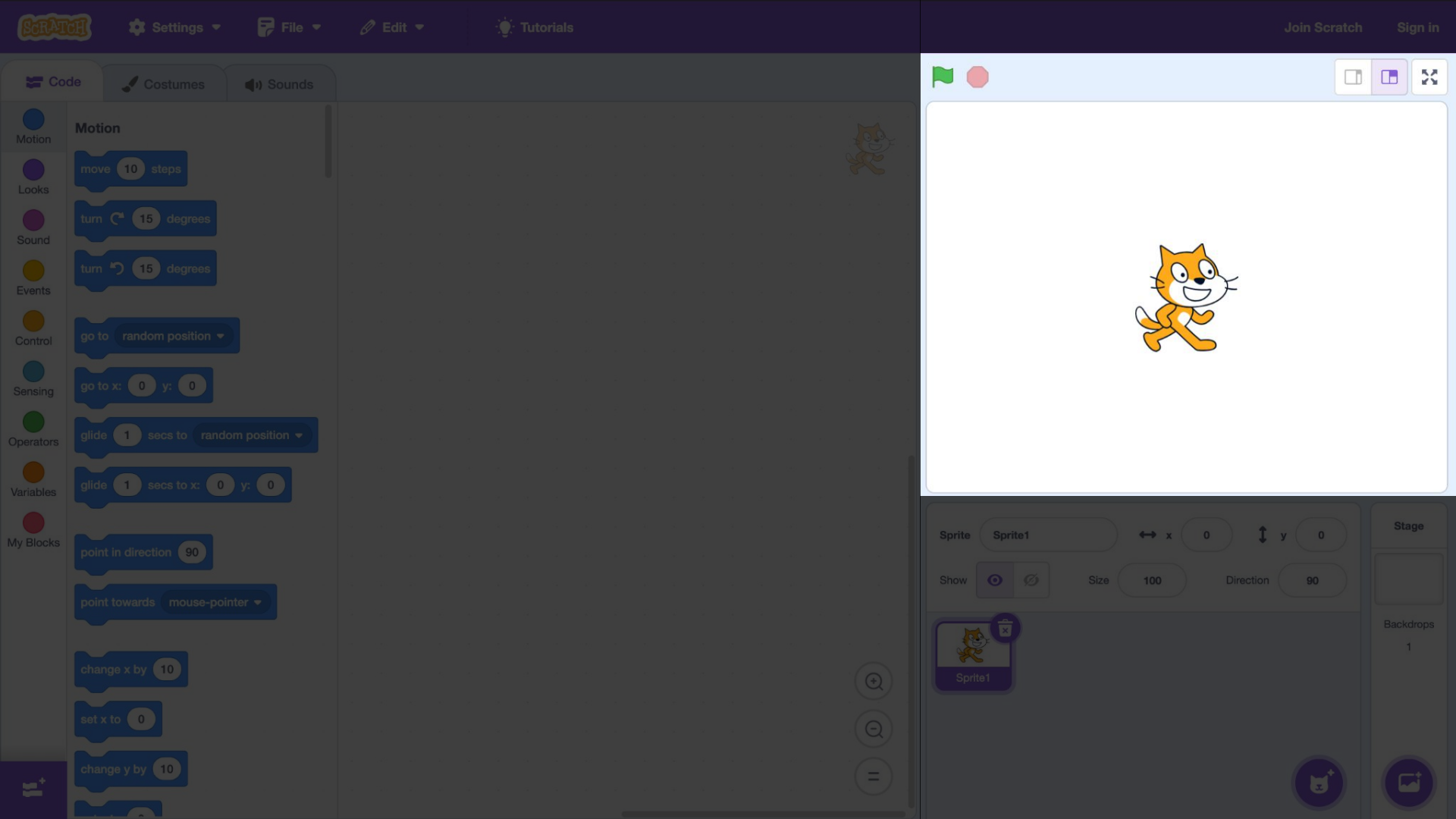
scratch.mit.edu

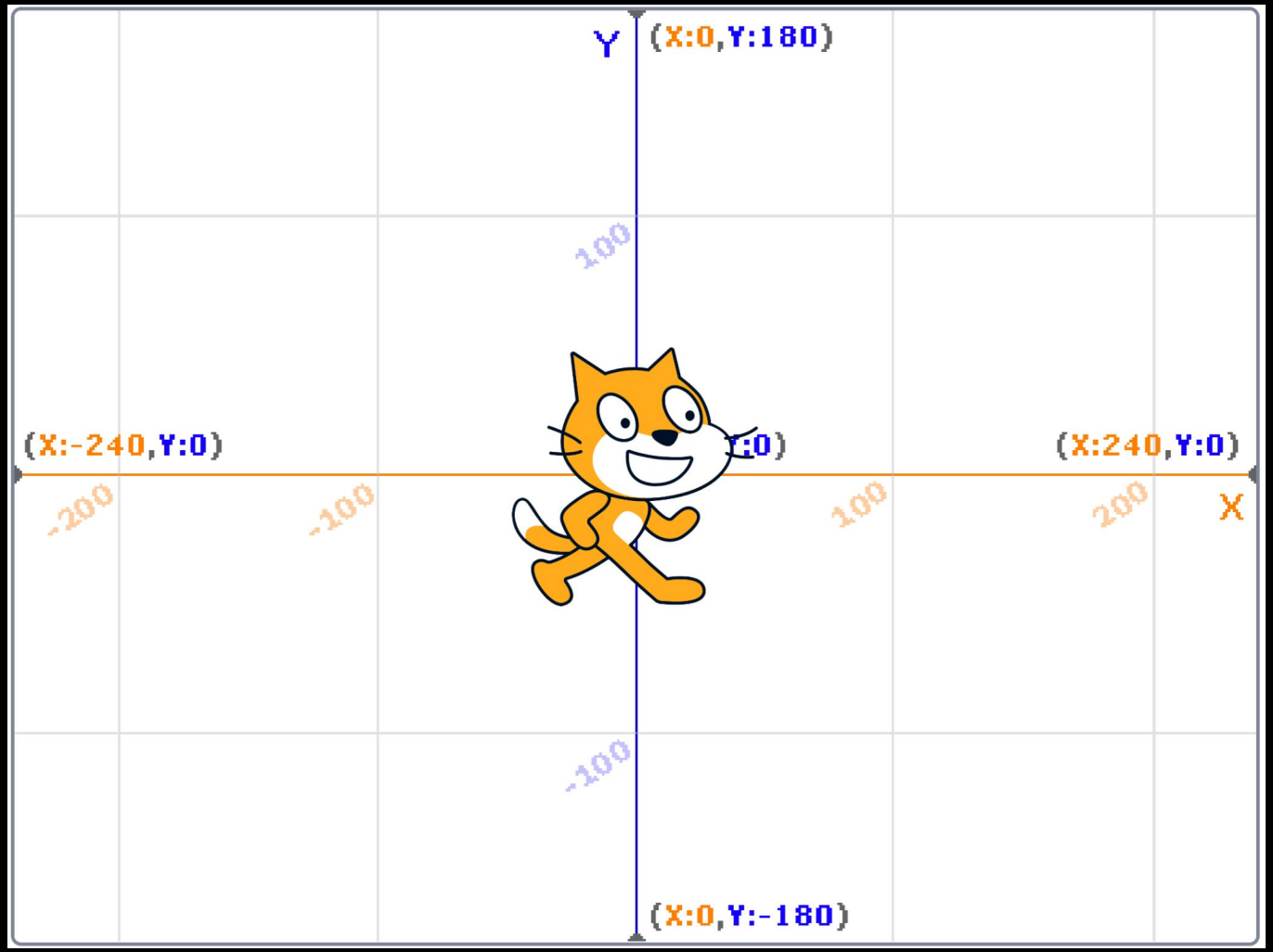

Settings
File
Edit
Tutorials
Join Scratch
Sign in
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
change y by 10
Sprite
Sprite1
x
0
y
0
Show
Size
100
Direction
90
Stage
Backdrops
1
Sprite1

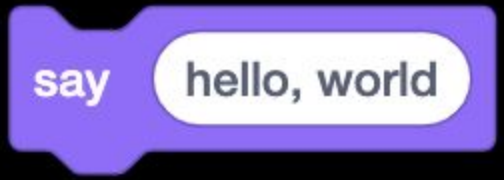

Settings
File
Edit
Tutorials
Join Scratch
Sign in
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
change y by 10
Sprite
Sprite1
x 0
y 0
Show
Size 100
Direction 90
Stage
Backdrops 1
Sprite1


Settings
File
Edit
Tutorials
Join Scratch
Sign in
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
change y by 10
Sprite
Sprite1
x 0
y 0
Show
Size 100
Direction 90
Stage
Backdrops 1
Sprite1

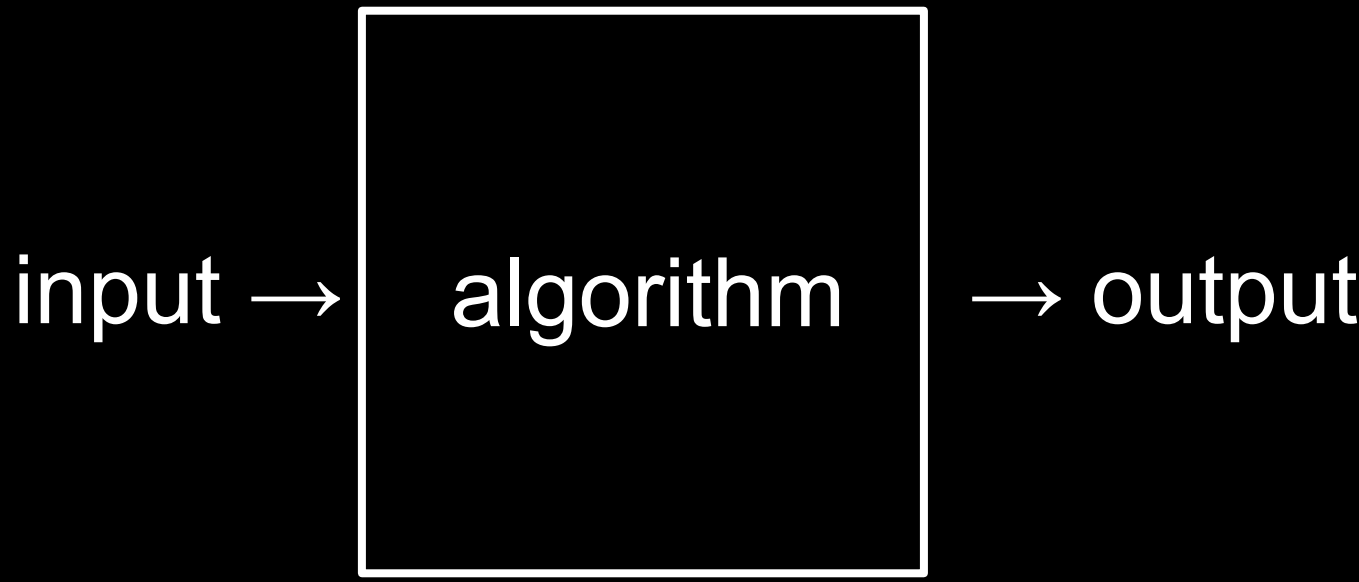

Settings
File
Edit
Tutorials
Join Scratch
Sign in
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
change y by 10
Sprite
Sprite1
x
0
y
0
Show
Size
100
Direction
90
Sprite1
Stage
Backdrops
1

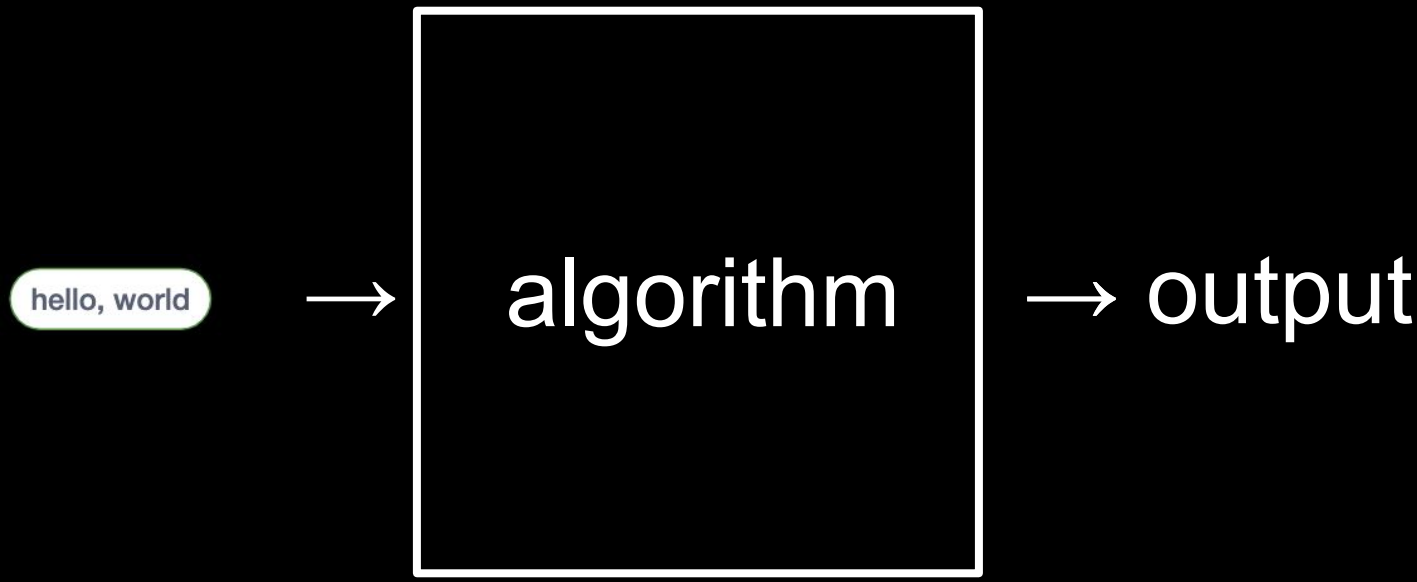

Settings
File
Edit
Tutorials
Join Scratch
Sign in
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
change y by 10
Sprite
Sprite1
x 0
y 0
Show
Size 100
Direction 90
Sprite1
Stage
Backdrops 1

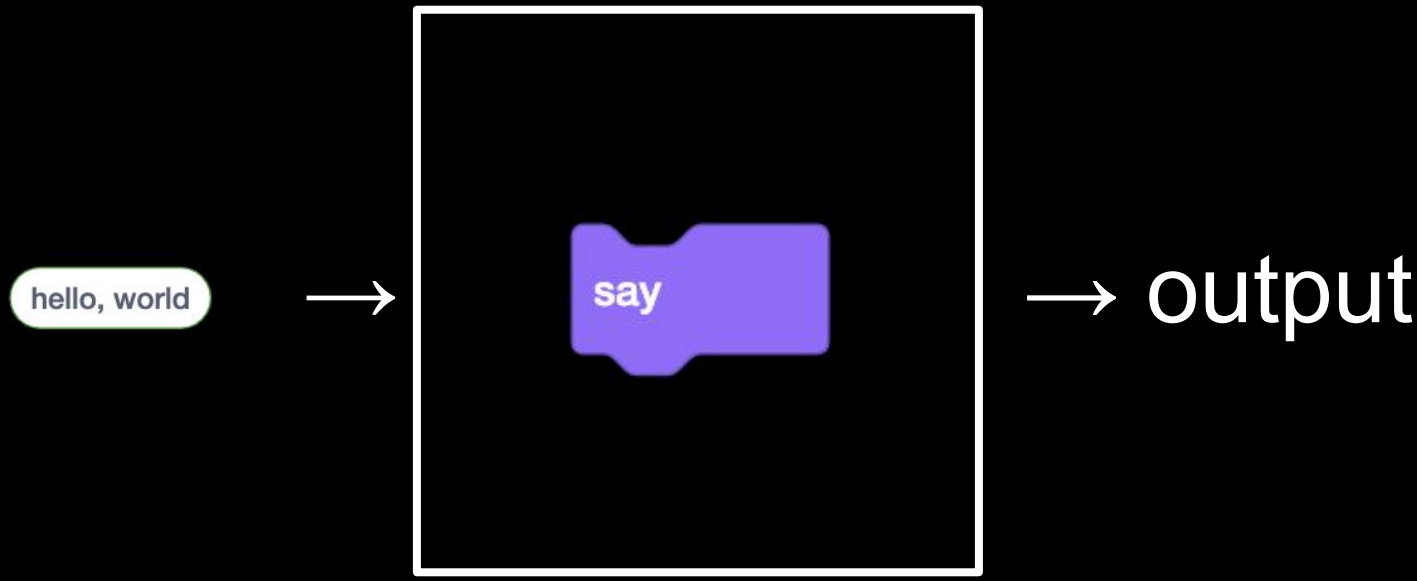

Y
(X:0,Y:180)
100
(X:-240,Y:0)
:0)
(X:240,Y:0)
-200
-100
100
200
X
-100
(X:0,Y:-180)

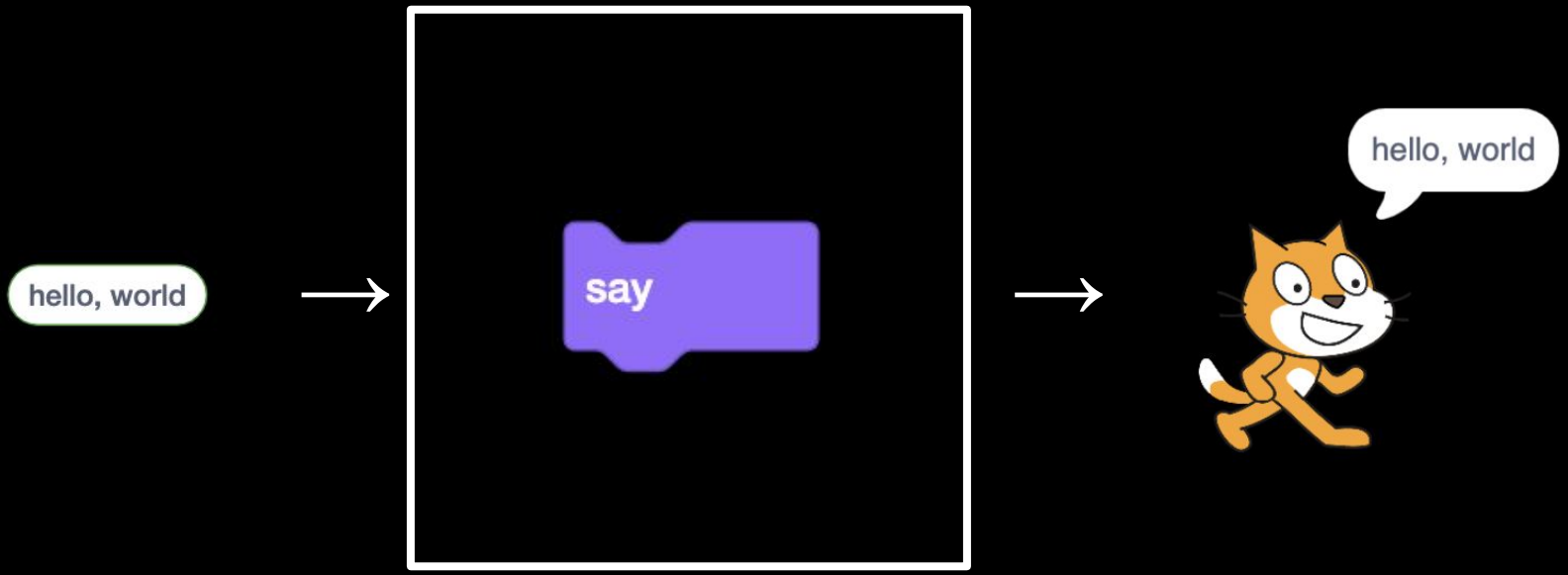

say
hello, world

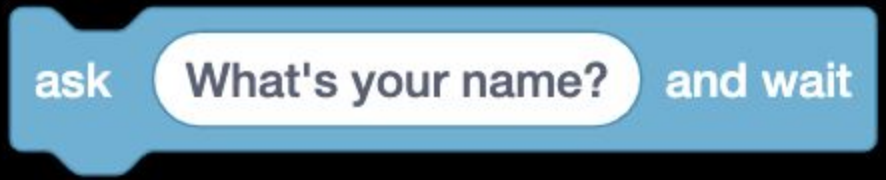
side effect


input →
algorithm
→ output

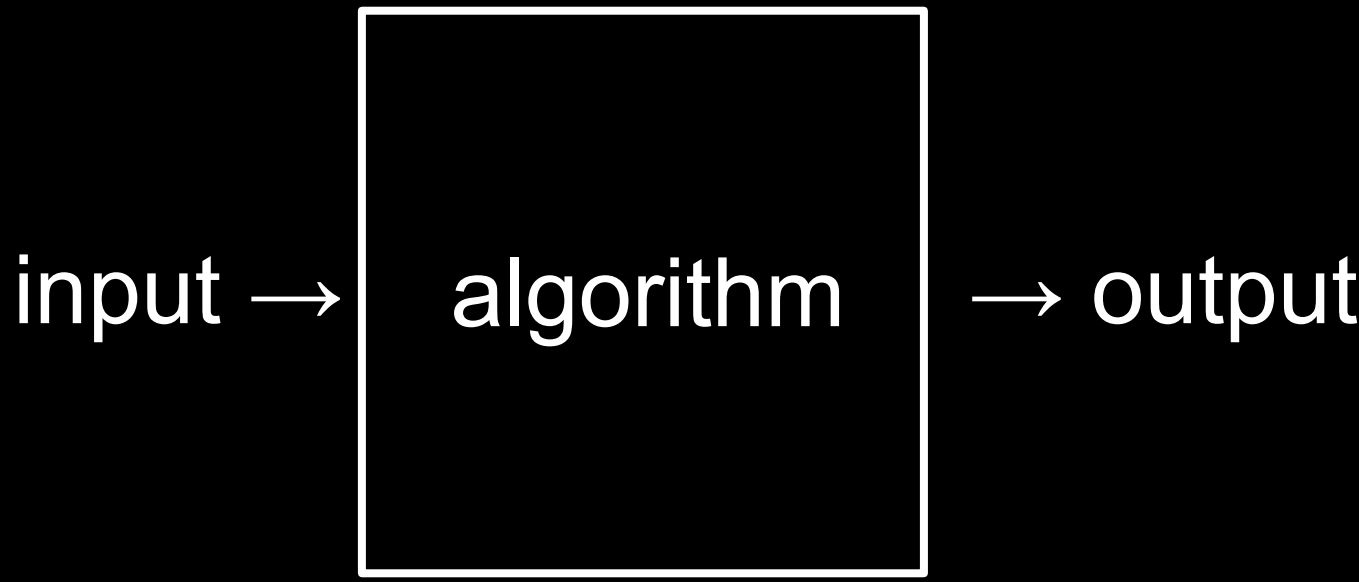

hello, world
algorithm
output

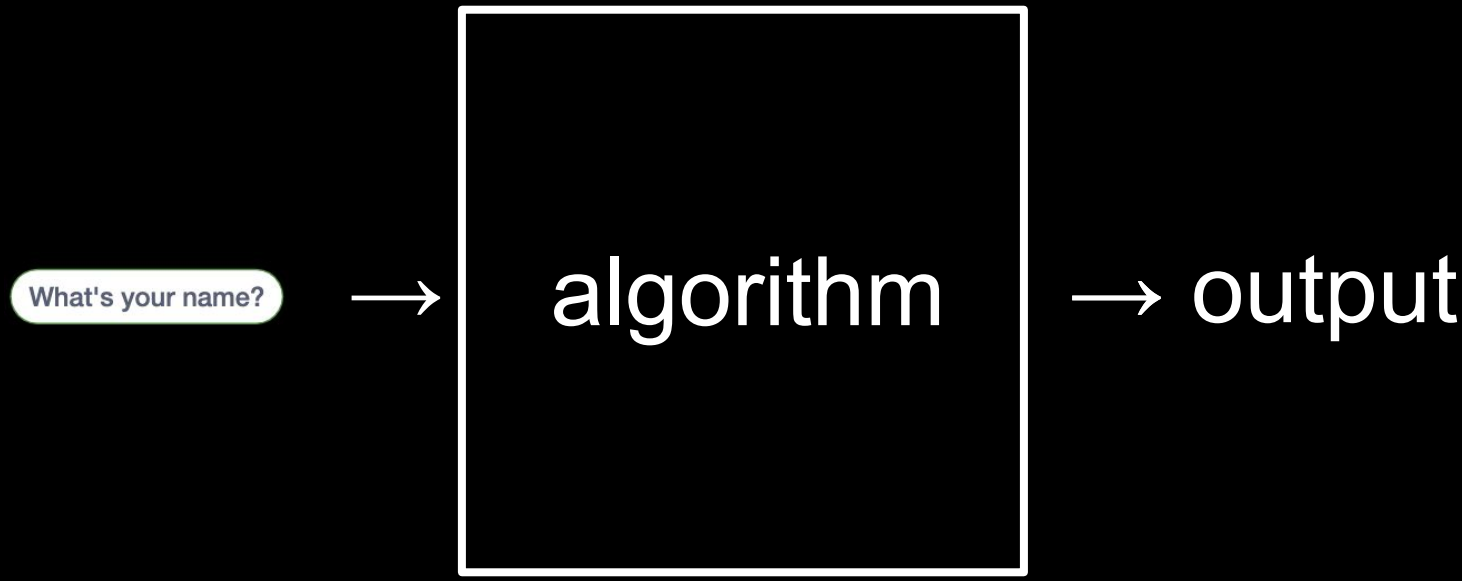

hello, world
say
output

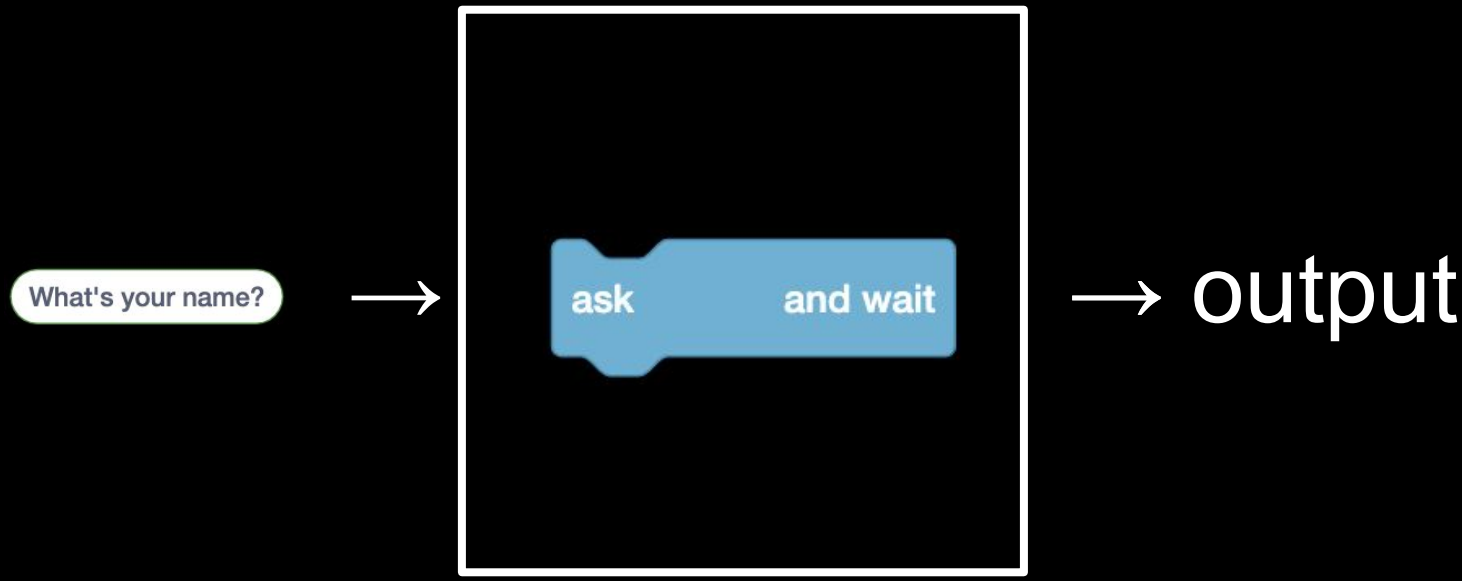

hello, world
say
hello, world

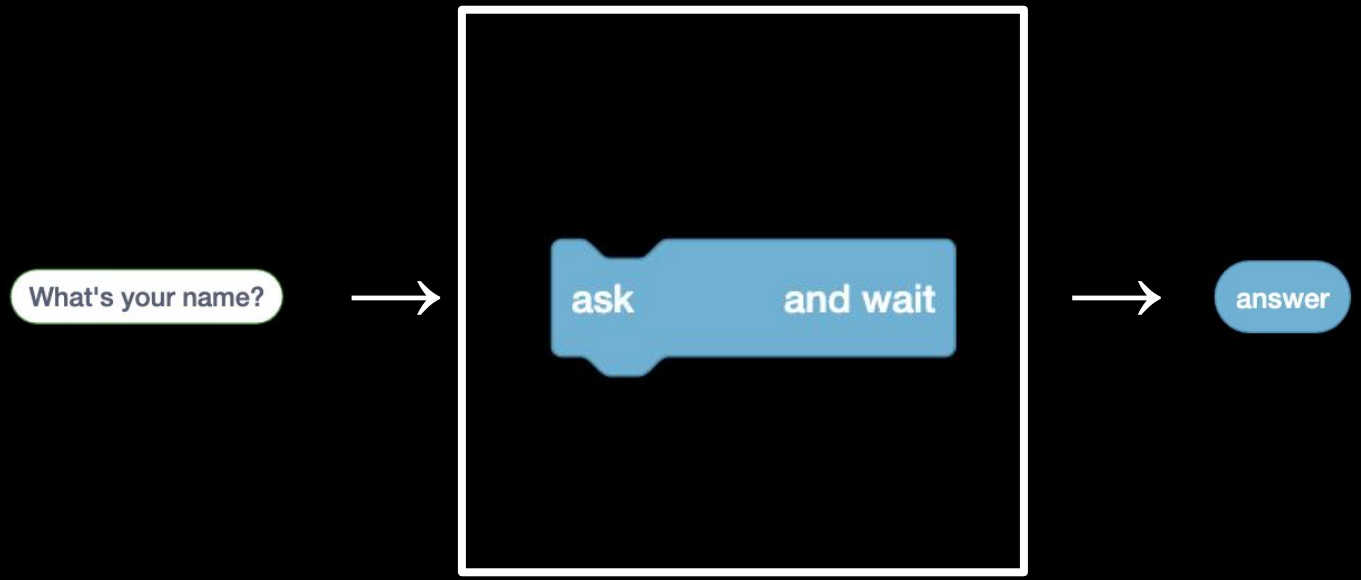

ask
What's your name?
and wait

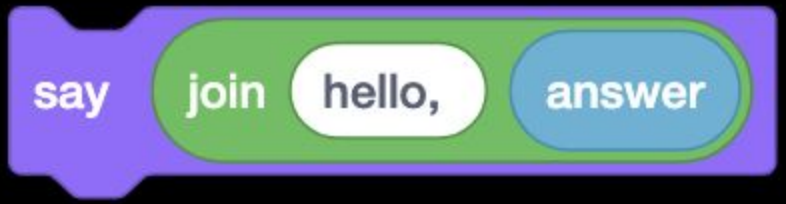
return value

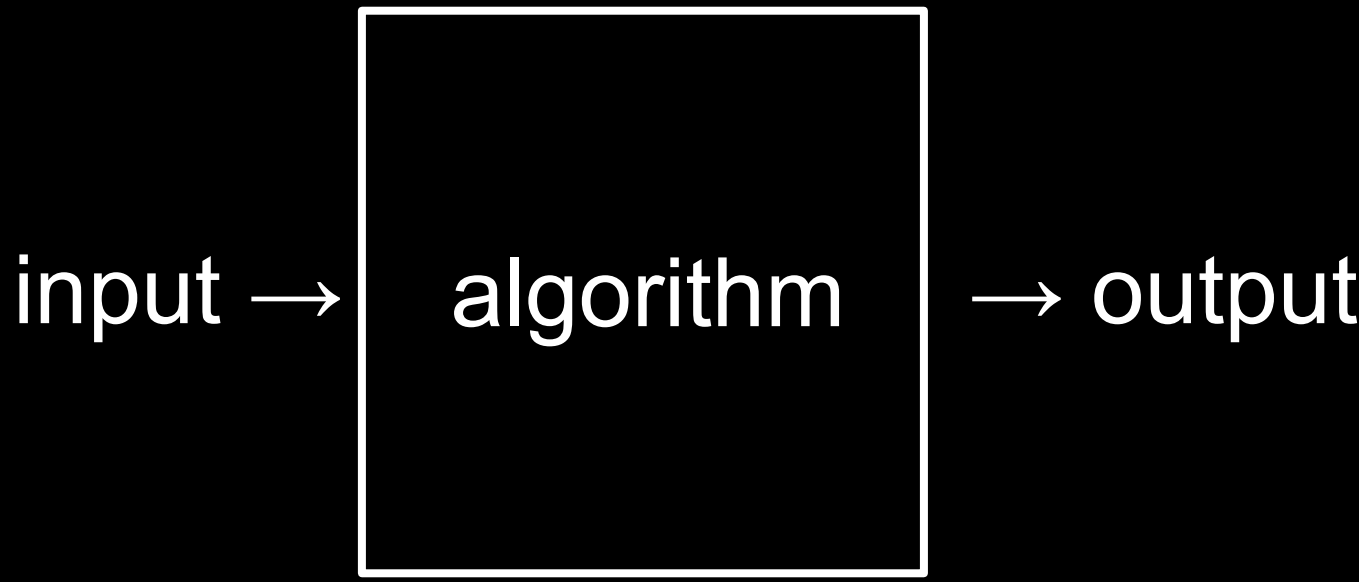

input →
algorithm
→ output

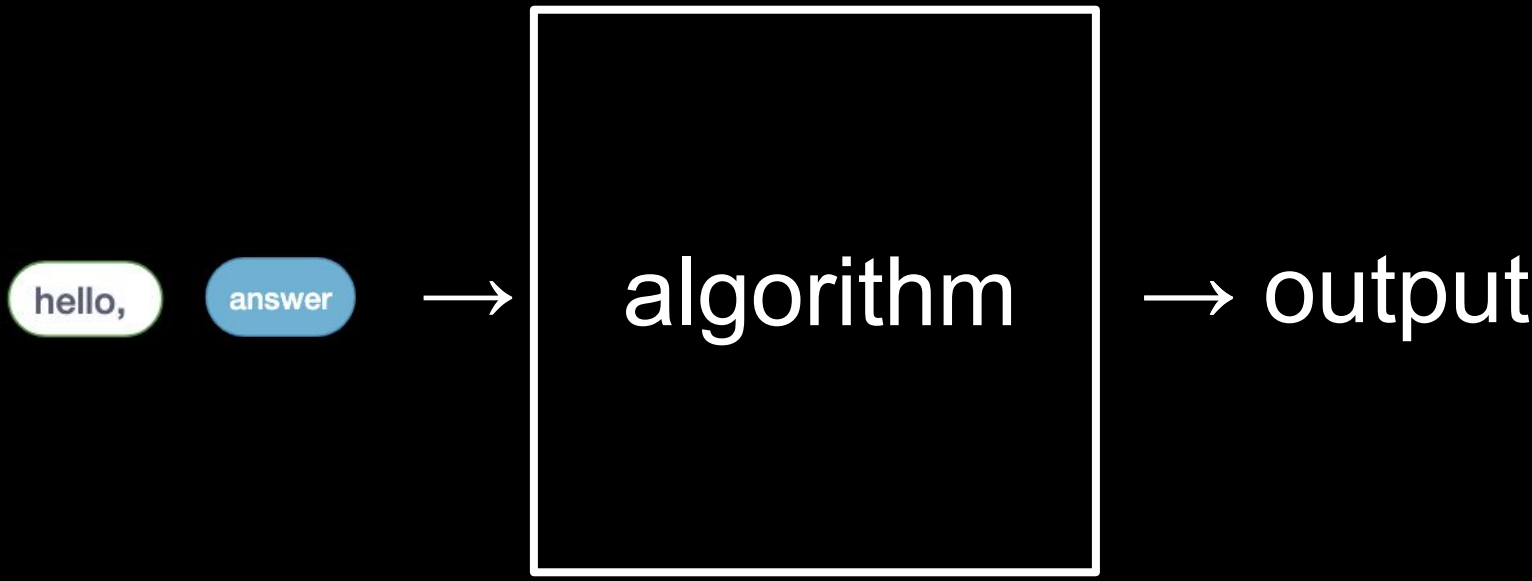

What's your name?
algorithm
output

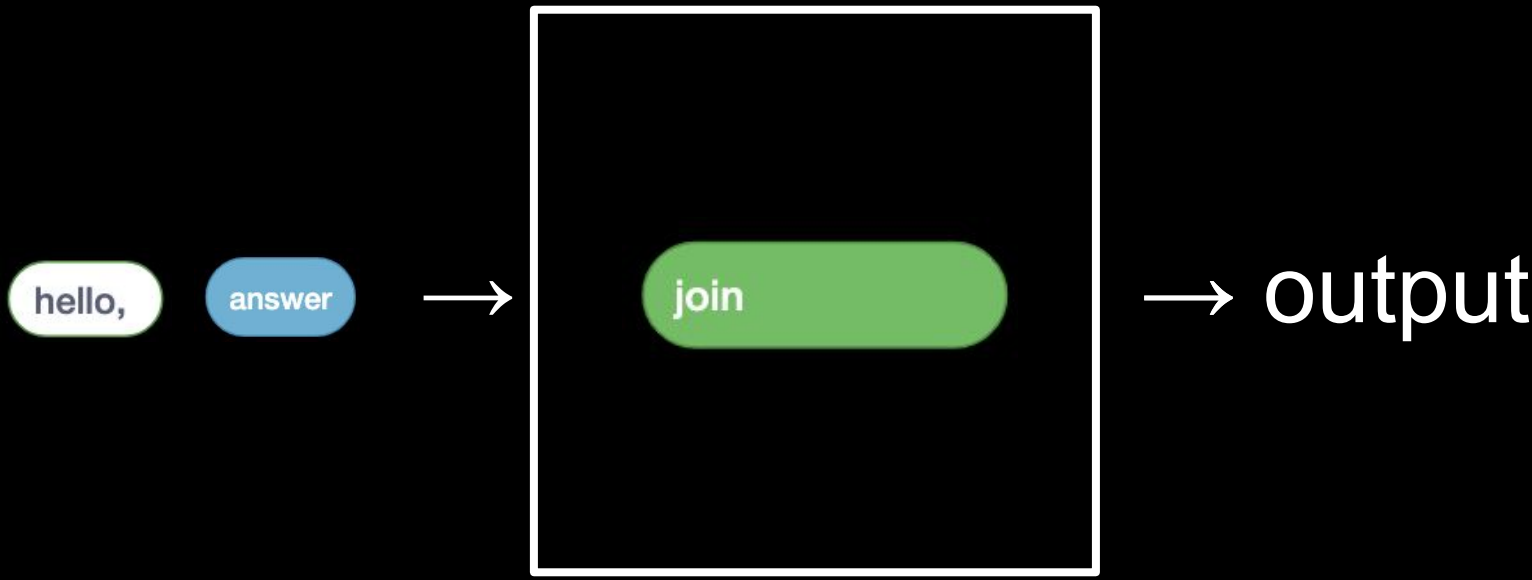

What's your name?
ask
and wait
output

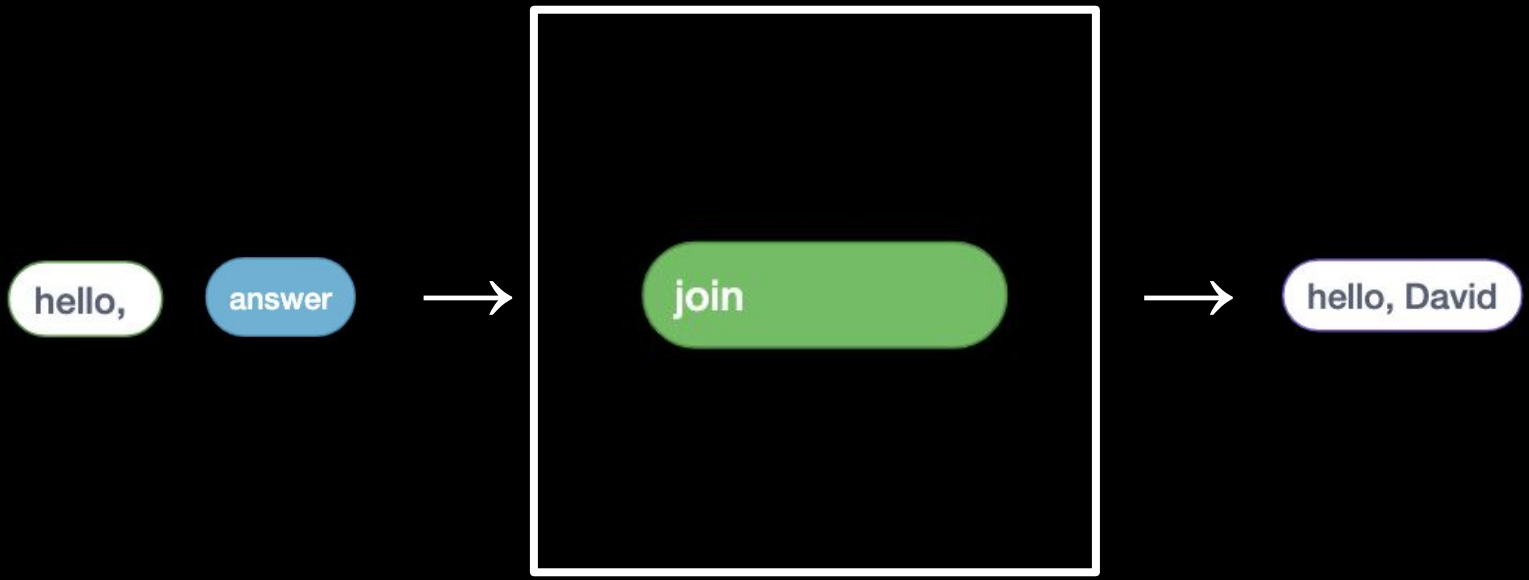

What's your name?
ask
and wait
answer


say
join
hello,
answer


input →
algorithm
→ output

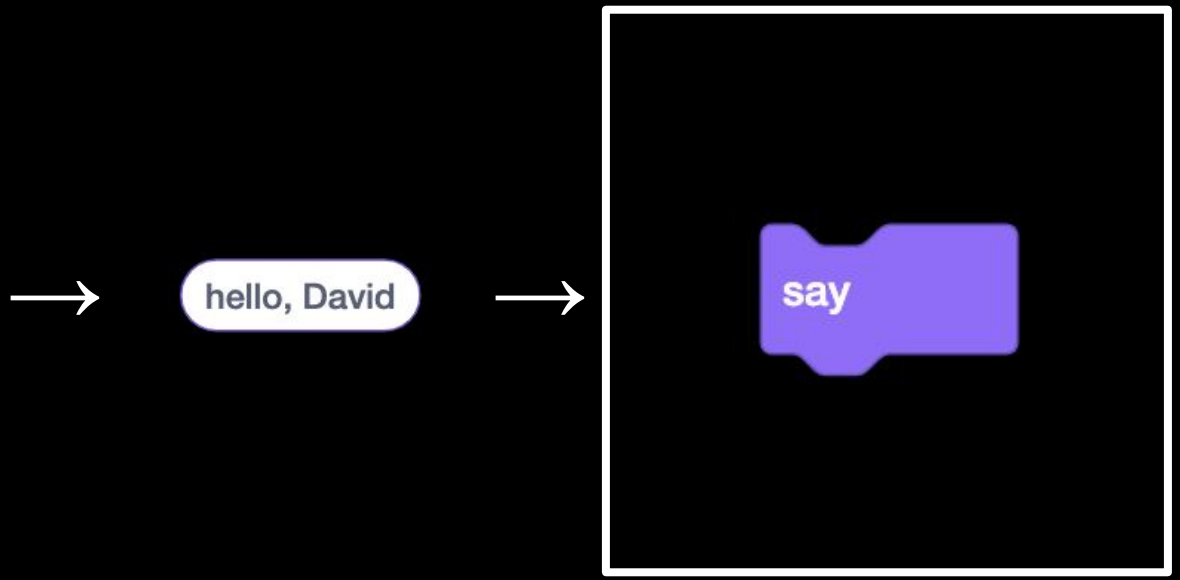

hello,
answer
algorithm
output


hello,
answer
join
output


hello,
answer
join
hello, David


hello, David


hello, David


hello, David
say


hello, David
say
hello, David

This is CS50