Hi

CAECO Friday

#### About Me

- Josh Pauli
- Dept Head & Professor at University of Arizona 2022-
- Professor & VP stuff at Dakota State University 2004-22
- Ph.D. in Software Engineering from NDSU
- Been with CAE-CO since inception in 2011
- Married w/ two adult daughters
- I like sports and math

#### Notations for Popular Number Sets

- We need to be comfortable with three main notations
  - Decimal = base 10, so you'll see 23<sub>10</sub> or just 23 like normal
  - Hex = 0x appears before the value, so you'll see 0x23 (zero-x)
  - Binary = 0b appears before the value. But it's only 0 and 1, so it's pretty obvious! (zero-b)
    - Most of the time, binary will just be a string of 0's and 1's without notation 0001110100010011

#### Character Sets

- The most common is ASCII
  - American Standard Code for Information Interchange
  - 128 accepted characters
  - Look at your keyboard how many keys? © (less than 128)

• This is the American standard. Other languages have HUGE character sets. Like Chinese. They use unicode.

# **ASCII Table**

Dec	Hex	0ct	Char	Dec	Hex	0ct	Char	Dec	Hex	0ct	Char	Dec	Hex	0ct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	Α	97	61	141	a
2	2	2		34	22	42		66	42	102	В	98	62	142	b
3	3	3		35	23	43	#	67	43	103	С	99	63	143	С
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47		71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	Н	104	68	150	h
9	9	11		41	29	51	)	73	49	111	1	105	69	151	i
10	Α	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	В	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	С	14		44	2C	54	,	76	4C	114	L	108	6C	154	1
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	Е	16		46	2E	56		78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	0	111	6F	157	0
16	10	20		48	30	60	0	80	50	120	Р	112	70	160	р
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	S
20	14	24		52	34	64	4	84	54	124	Т	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	V
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	X
25	19	31		57	39	71	9	89	59	131	Υ	121	79	171	У
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	

### **ASCII Table**

Dec	Hex	0ct	Char	Dec	Hex	0ct	Char	Dec	Hex	0ct	Char	Dec	Hex	0ct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	Α	97	61	141	a
2	2	2		34	22	42	"	66	42	102	В	98	62	142	b
3	3	3		35	23	43	#	67	43	103	С	99	63	143	С
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	•	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	Н	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	Α	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	В	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	С	14		44	2C	54	,	76	4C	114	L	108	6C	154	I
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56		78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	0	111	6F	157	0
16	10	20		48	30	60	0	80	50	120	Р	112	70	160	р
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	S
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	V
23	17	27		55	37	67	7	87	57	127	W	119	77	167	W
24	18	30		56	38	70	8	88	58	130	X	120	78	170	X
25	19	31		57	39	71	9	89	59	131	Υ	121	79	171	У
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	Z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	

### **ASCII Table**

Dec	Hex	0ct	Char	Dec	Hex	0ct	Char	Dec	Hex	0ct	Char	Dec	Hex	0ct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	Α	97	61	141	а
2	2	2		34	22	42	"	66	42	102	В	98	62	142	b
3	3	3		35	23	43	#	67	43	103	С	99	63	143	С
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	•	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	Н	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	Α	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	В	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	С	14		44	2C	54	,	76	4C	114	L	108	6C	154	1
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	Е	16		46	2E	56		78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	0	111	6F	157	0
16	10	20		48	30	60	0	80	50	120	Р	112	70	160	р
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	S
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	V
23	17	27		55	37	67	7	87	57	127	W	119	77	167	W
24	18	30		56	38	70	8	88	58	130	X	120	78	170	X
25	19	31		57	39	71	9	89	59	131	Υ	121	79	171	У
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	Z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137		127	7F	177	

# Converting Decimal to Binary

### Decimal to Binary

- Base<sub>10</sub> to Base<sub>2</sub>
- What is 81 in binary? (nothing but 0s and 1s)
  - How about 187?
  - How about 44,503?

- Really easy. Just divide by 2 until you can't anymore!
- "Repeated divide by 2"

# Decimal 81 = binary?

	Whole Number	Remainder
81/2	40	

	Whole Number	Remainder
81/2	40	
	Rinse and	
	Rinse and Repeat!	

	Whole Number	Remainder
81/2	-40	
40 / 2◀	20	0

	Whole Nur	mber Remainder	
81/2	40		
40 / 2 ←	20	0	
20 / 2	10	0	

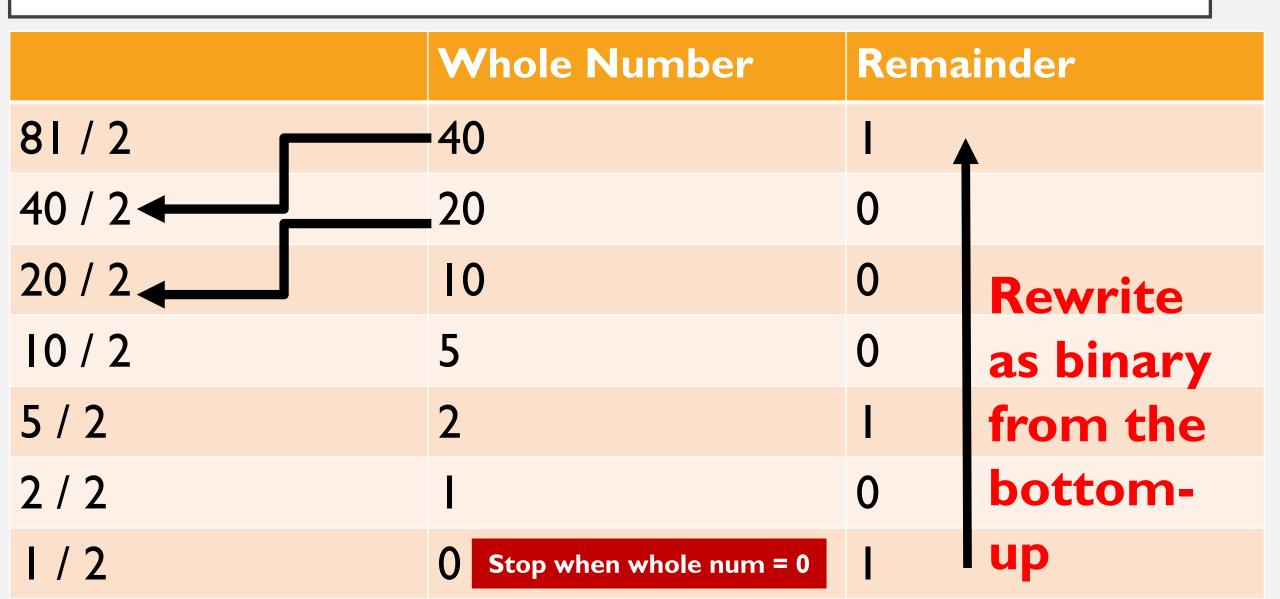
	Whole Number	Remainder
81/2	40	
40 / 2 ←	20	0
20 / 2	10	0
10/2	5	0

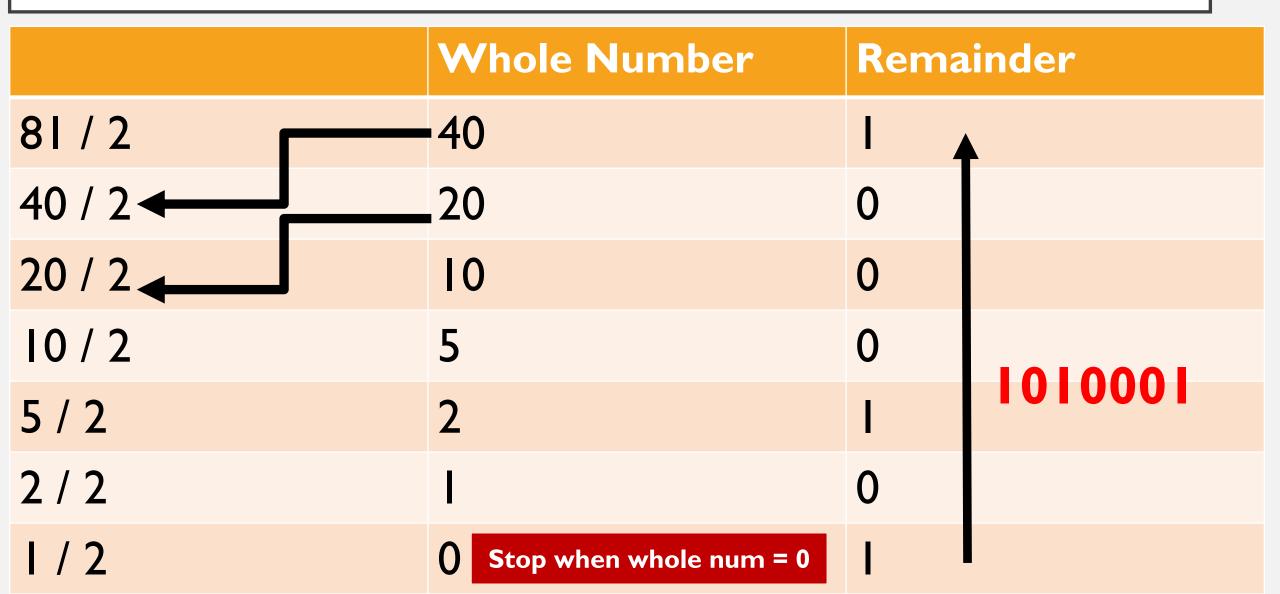
	Whole Number	Remainder
81/2	<b>-</b> 40	
40 / 2 ←	_20	0
20 / 2	10	0
10/2	5	0
5 / 2	2	

	Whole Number	Remainder
81/2	<b>-</b> 40	
40 / 2 ←	_20	0
20 / 2	10	0
10/2	5	0
5 / 2	2	Í
2/2		0

	Whole Number	Remainder
81/2	-40	Ī
40 / 2 ←	_20	0
20 / 2	10	0
10/2	5	0
5 / 2	2	
2/2		0
1/2	0	

	Whole Number	Remainder
81/2	<b>-40</b>	
40 / 2 ←	_20	0
20 / 2	10	0
10 / 2	5	0
5 / 2	2	
2/2	I	0
1/2	O Stop when whole num = 0	

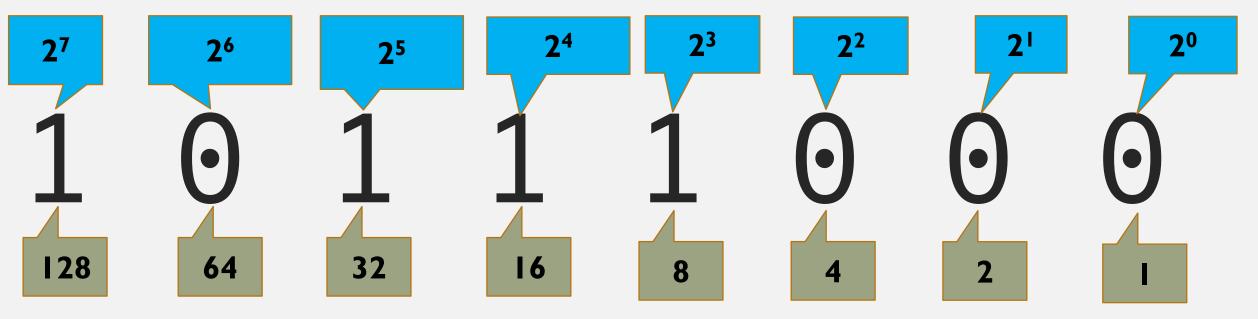




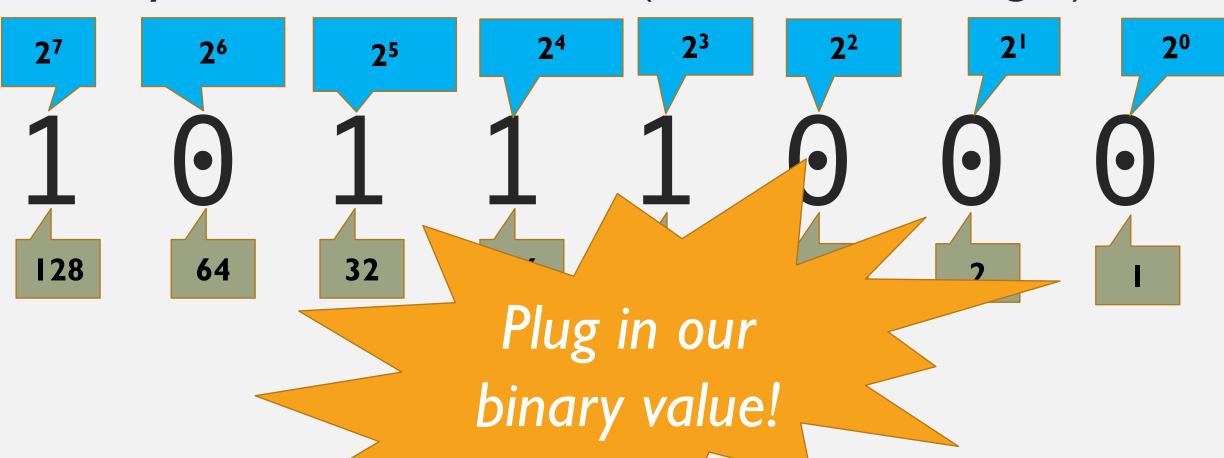
# Is 81 really 1010001?

• We can check that, too!

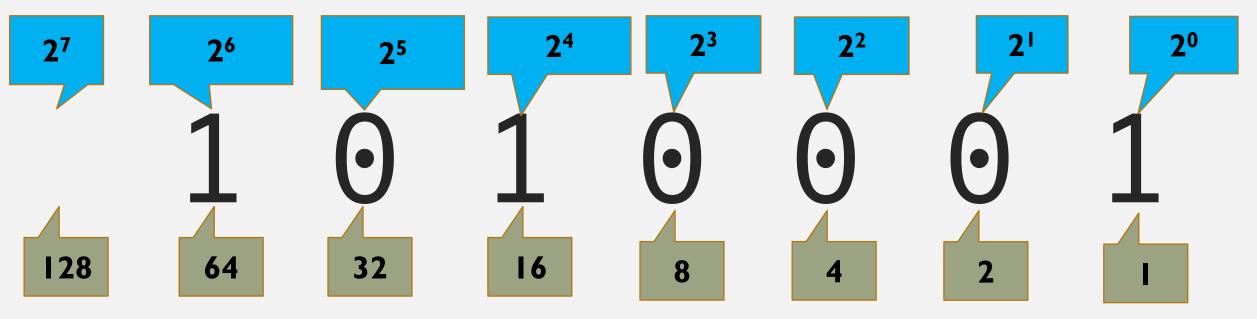
#### Remember this?



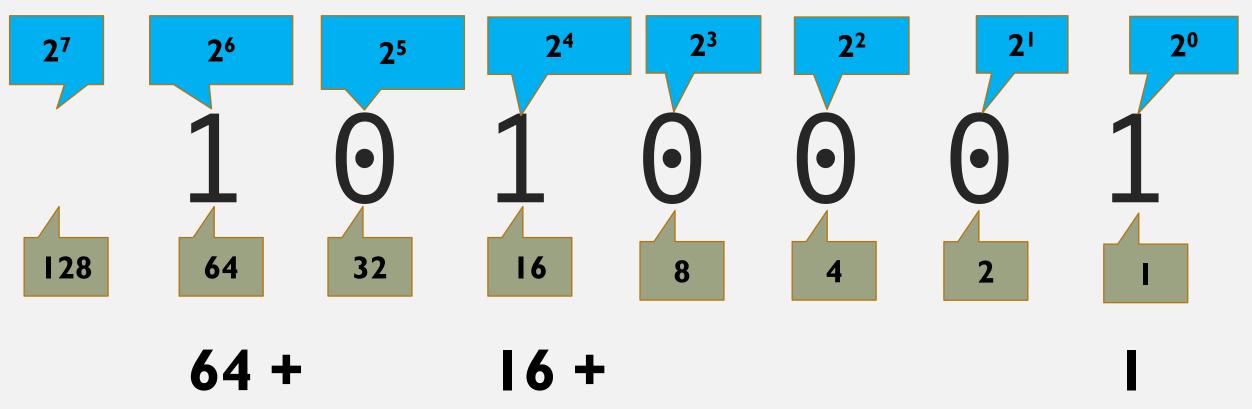
#### Remember this?



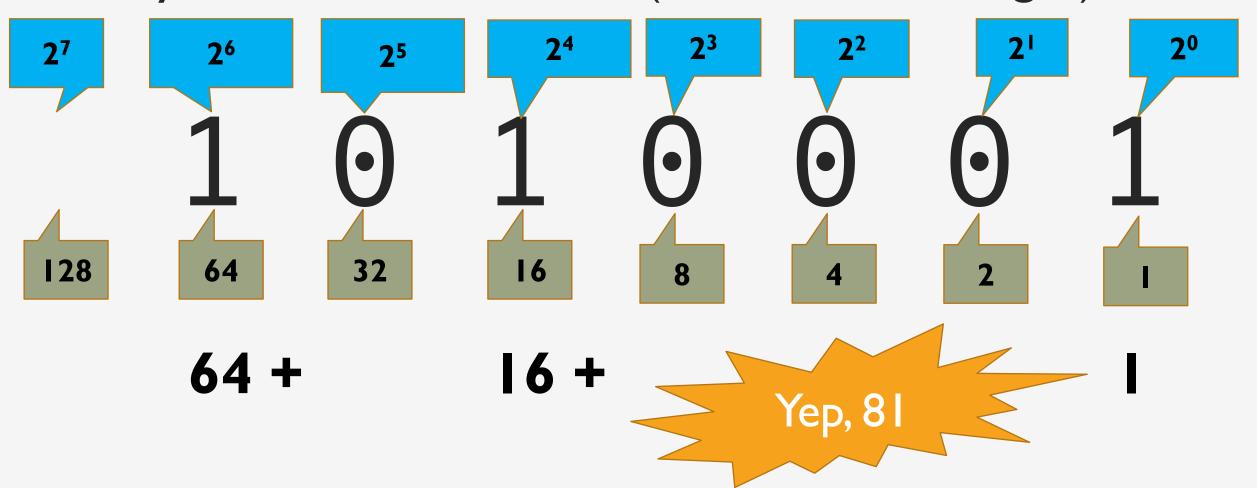
#### Here's our binary for 81 - is it correct?



### Here's our binary for 81. Is it correct?



#### Here's our binary for 81. Is it correct?



	Whole Number	Remainder
187 / 2		

	Whole Number	Remainder
187 / 2	93	

	Whole Number	Remainder
187 / 2	93	
93 / 2	46	

	Whole Number	Remainder
187 / 2	93	
93 / 2	46	
46 / 2	23	0

	Whole Number	Remainder
187 / 2	93	
93 / 2	46	
46 / 2	23	0
23 / 2	11	

	Whole Number	Remainder
187 / 2	93	
93 / 2	46	
46 / 2	23	0
23 / 2		
11/2	5	

	Whole Number	Remainder
187 / 2	93	
93 / 2	46	
46 / 2	23	0
23 / 2		
11/2	5	
5 / 2	2	

	Whole Number	Remainder
187 / 2	93	
93 / 2	46	
46 / 2	23	0
23 / 2	11	
11/2	5	
5 / 2	2	
2/2		0

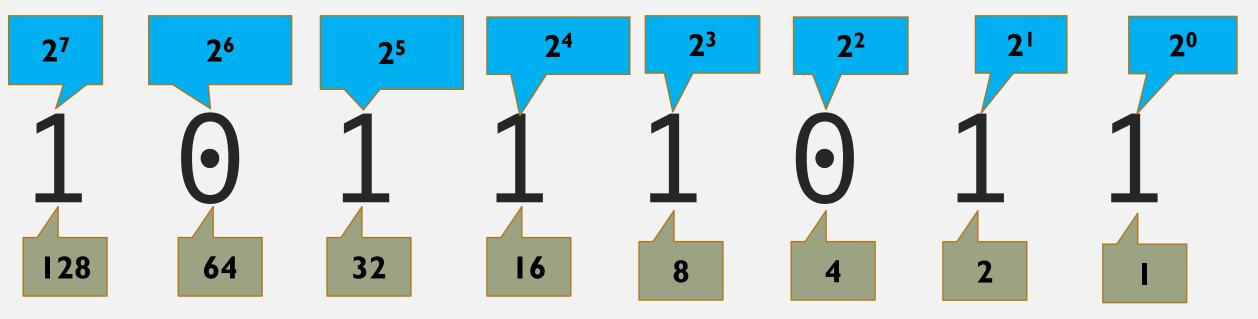
	Whole Number	Remainder
187 / 2	93	I
93 / 2	46	
46 / 2	23	0
23 / 2	11	1
11/2	5	
5 / 2	2	
2/2		0
I / 2	0	I

## Let's do it again! Repeated Divide by 2

	Whole Number	Rema	inder
187 / 2	93	I	
93 / 2	46	ı	
46 / 2	23	0	
23 / 2		I	
11/2	5	I	10111011
5 / 2	2	I	
2/2	1	0	
I / 2	0	I	

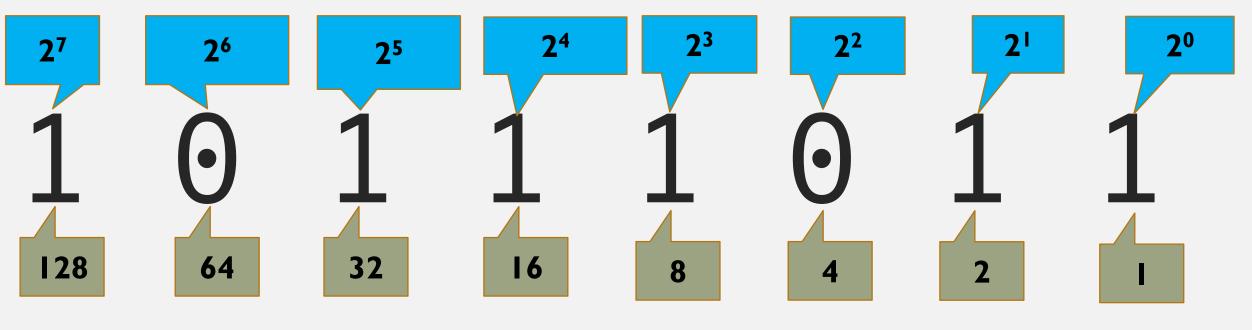
### Here's our binary for 187 - is it correct?

Binary uses base2 numbers (start from the right)



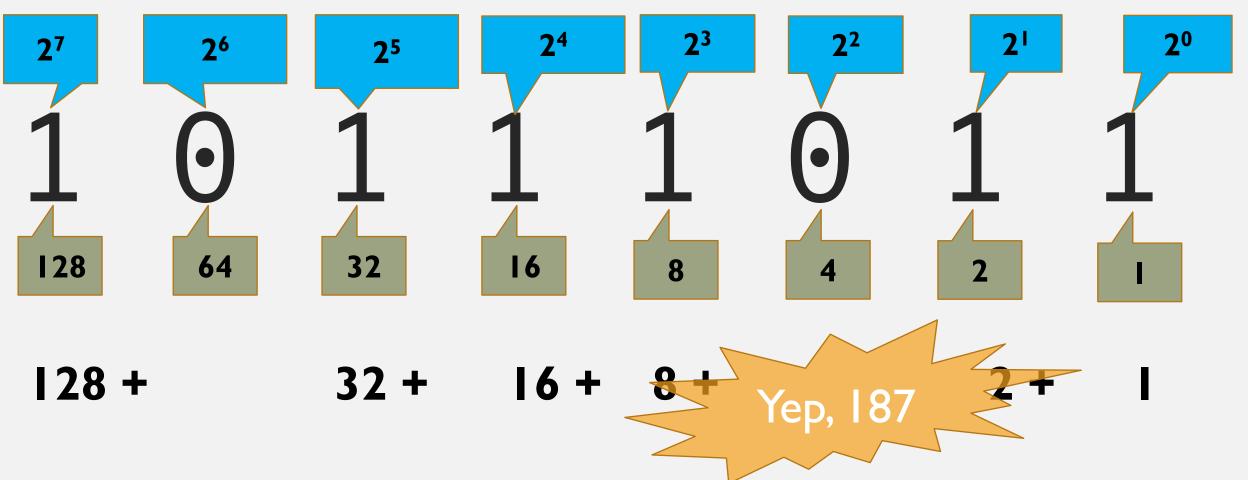
### Here's our binary for 187 - is it correct?

Binary uses base2 numbers (start from the right)



### Here's our binary for 187 - is it correct?

Binary uses base2 numbers (start from the right)



### Converting Decimal to Binary

Big numbers take a long time, but same process!

- How about converting Binary to Decimal?
  - We did it to check our answers! ©

### Converting Decimal to Hex

### Decimal to Hex

- Base<sub>10</sub> to Base<sub>16</sub>
- Remember the hex character set?
  - 0-9, A-F
  - 16 total characters

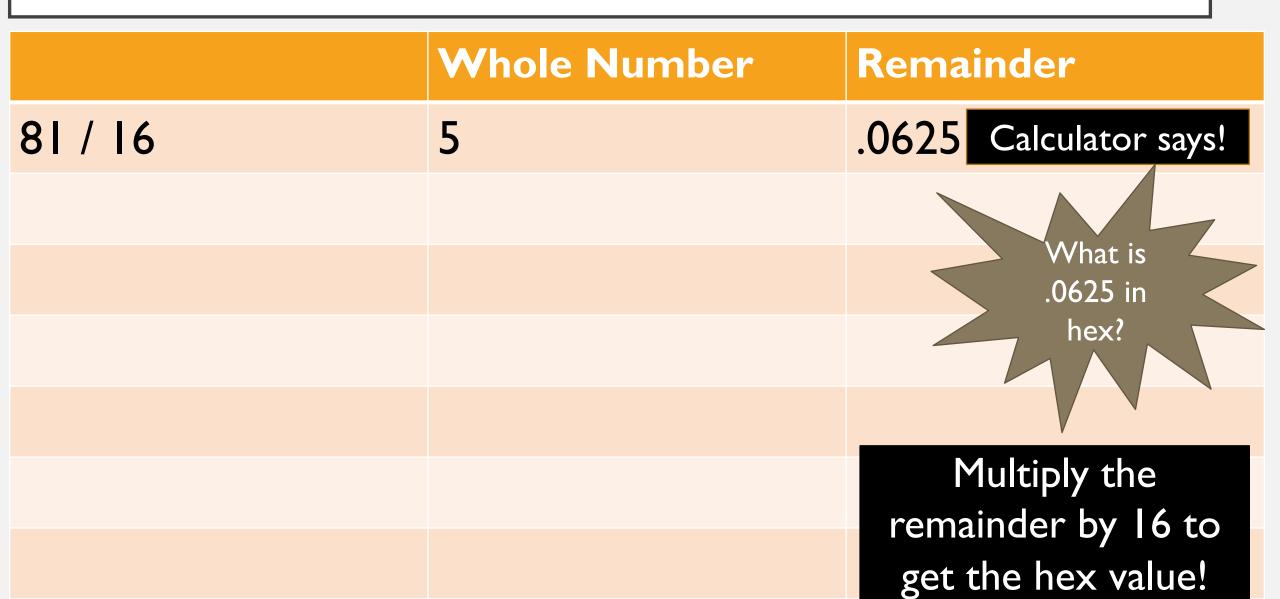
- Really easy. Just divide by 16 until you can't anymore!
- "Repeated divide by 16"

## Decimal 81 = hex?

	Whole Number	Remainder
81 / 16	5	.0625

	Whole Number	Remainder
81/16	5	.0625 Calculator says!

	Whole Number	Remainder
81 / 16	5	.0625 Calculator says!
		What is .0625 in hex?
		hex?

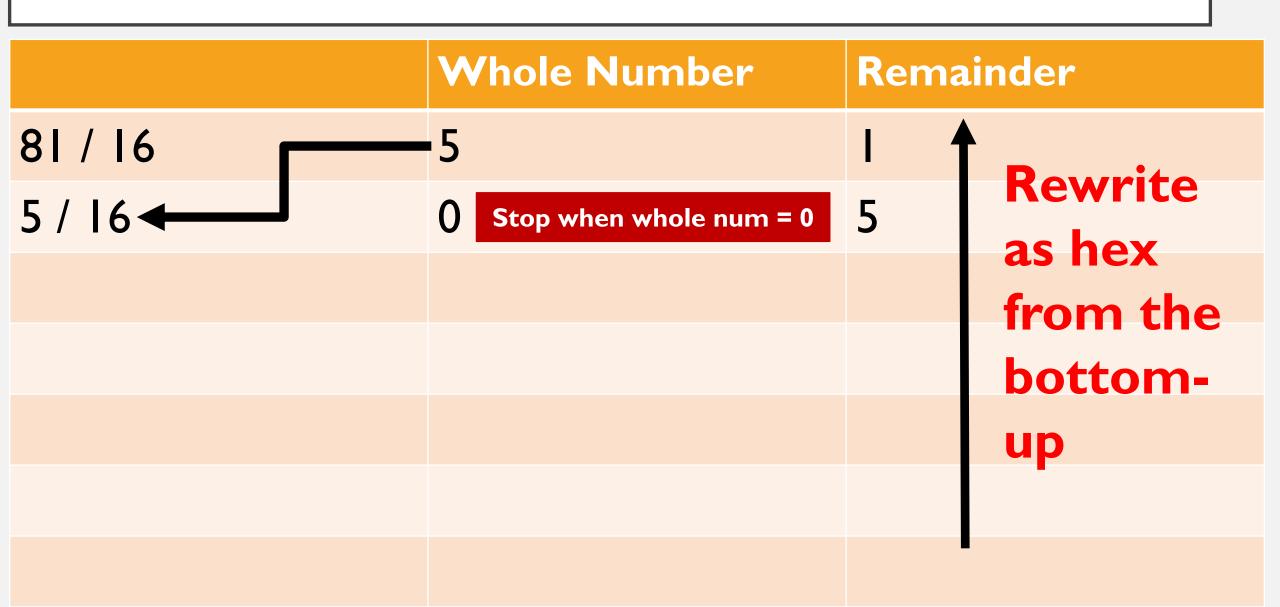


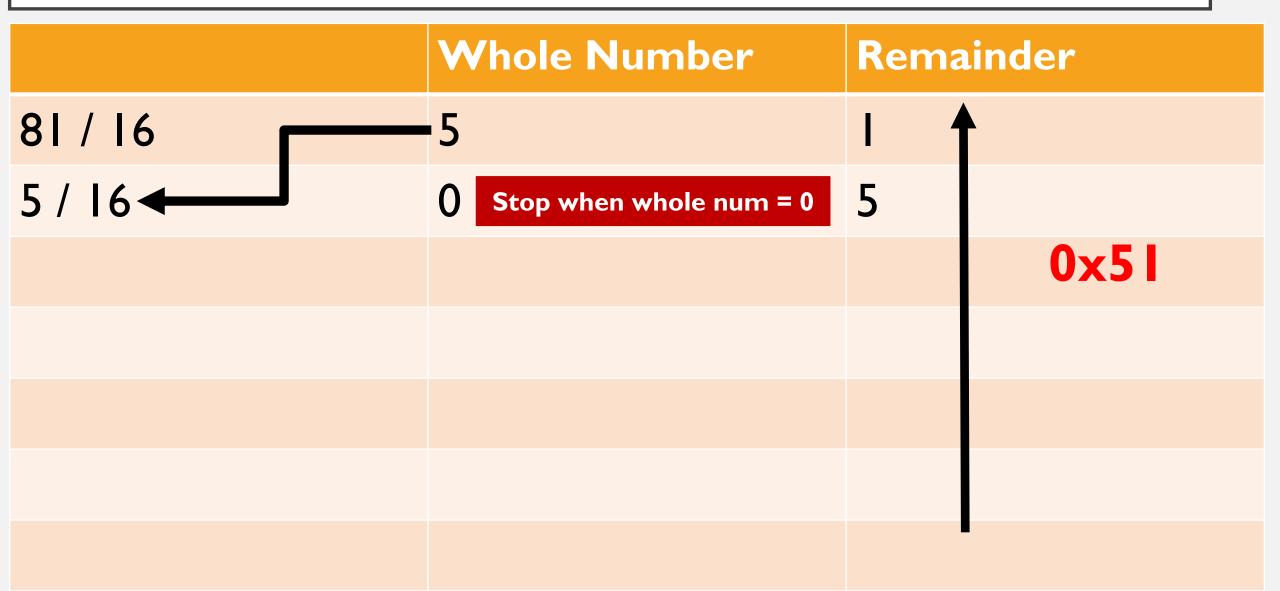
	Whole Number	Remainder
81/16	5	$.0625 \times 16 = 1$

	Whole Number	Rema	inder
81/16	5	I	$.0625 \times 16 = 1$
	Rinse and		
	Repeat!		

	Whole Number	Remainder
81 / 16 5 / 16 <del>←</del>	-5	
5 / 16 ←	0	5

	Whole Number	Remainder
81 / 16	-5	
5 / 16 ←	O Stop when whole num = 0	5





### Is 81 really 0x51?

• We can check that, too!

• But no spoilers yet!

We'll do hex-to-decimal conversion next!

# Decimal 187 = hex?

	Whole Number	Remainder
187 / 16		.6875

	Whole Number	Remainder	
187 / 16	11	.6875 Calculator says!	

	Whole Number	Remainder
187 / 16		.6875 Calculator says!
		Multiply the remainder by 16 to
		get the hex value!

60

	Whole Number	Remainder
187 / 16	11	.6875 × 16 = 11

	Whole Number	Remainder
187 / 16	ÍI	.6875 .6875 x 16 = 11
		Is II a hex
		character?

### Hex Character Set

Decimal	Hex
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7

Decimal	Hex
8	8
9	9
10	Α
11	В
12	С
13	D
14	E
15	F

### Hex Character Set

Decimal	Hex
0	0
I	1
2	2
3	3
4	4
5	5
6	6
7	7

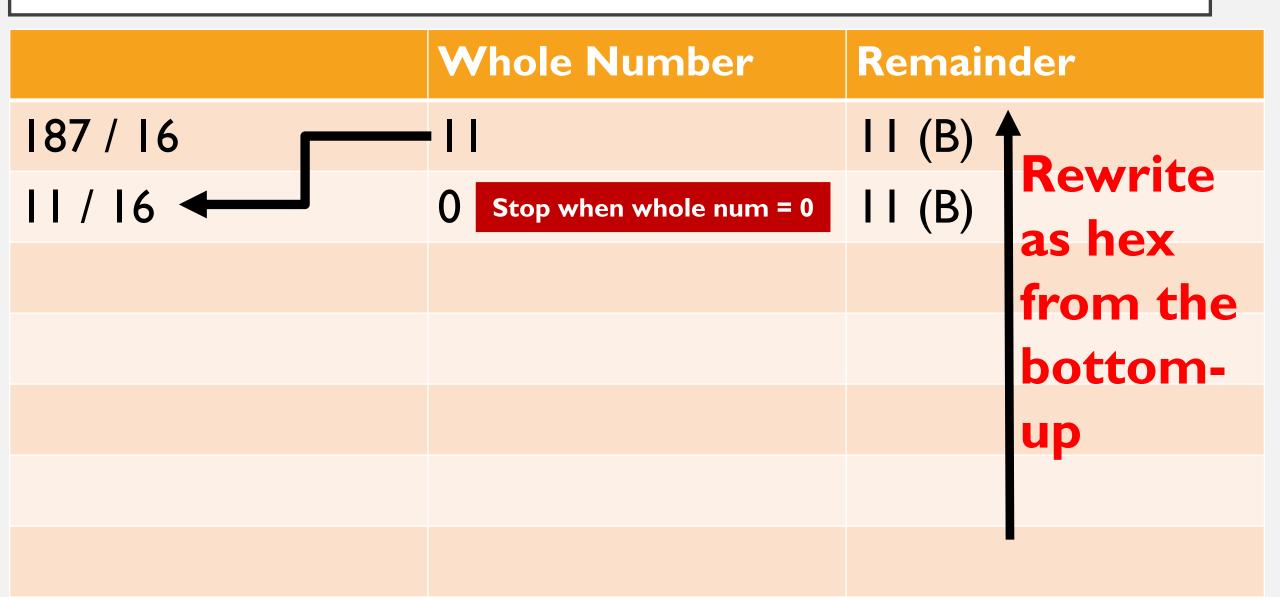
Decimal	Hex	
8	8	
9	9	
10	Α	
11	В	
12	С	
13	D	
14	E	
15	F	

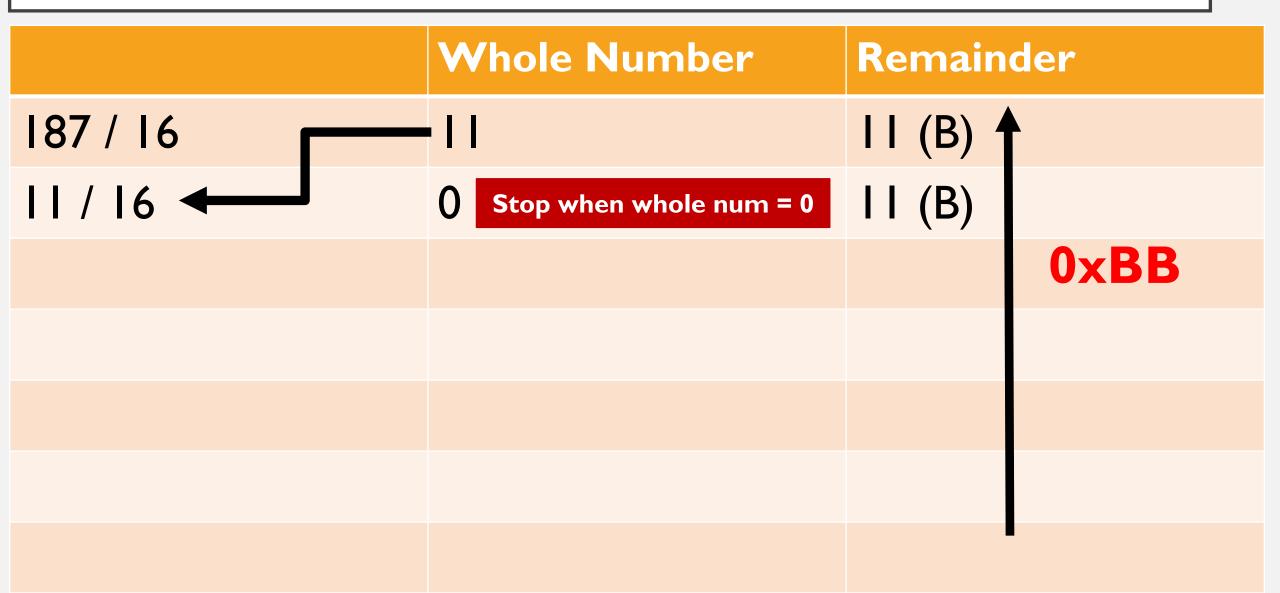
#### 64

	Whole Number	Remainder
187 / 16		II (B)

	Whole Number	Remainder
187 / 16	-	II (B)
187 / 16 11 / 16	0	II (B)

	Whole Number	Remainder
187 / 16	-	II (B)
11 / 16	O Stop when whole num = 0	II (B)





### Is 81 really 0x51? Is 187 really 0xBB?

• Two to check!

• Let's do a BIG number!

	Whole Number	Remainder
58,345 / 16		

	Whole Number	Remainder
58,345 / 16	3,646	.5625

	Whole Number	Remainder
58,345 / 16	3,646	9

	Whole Number	Remainder
58,345 / 16	3,646	9
3,646 / 16	227	14 (E)

	Whole Number	Remainder
58,345 / 16	3,646	9
3,646 / 16	227	14 (E)
227 / 16	14	3

	Whole Number	Remainder
58,345 / 16	3,646	9
3,646 / 16	227	14 (E)
227 / 16	14	3
14/16	0	14 (E)

	Whole Number	Remainder
58,345 / 16	3,646	9
3,646 / 16	227	14 (E)
227 / 16	14	3
14/16	O Stop when whole num = 0	14 (E)

	Whole Number	Remain	der
58,345 / 16	3,646	9	
3,646 / 16	227	14 (E)	
227 / 16	14	3	
14/16	O Stop when whole num = 0	14 (E)	
			0xE3E9

#### Need to check these...

- Three to check!
  - Is 81 really 0x51?
  - Is 187 really 0xBB?
  - Is 58,345 really 0xE3E9?

• The key is just to understand what power of 2 you're dealing with.

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	I

• The key is just to understand what power of 2 you're dealing with.

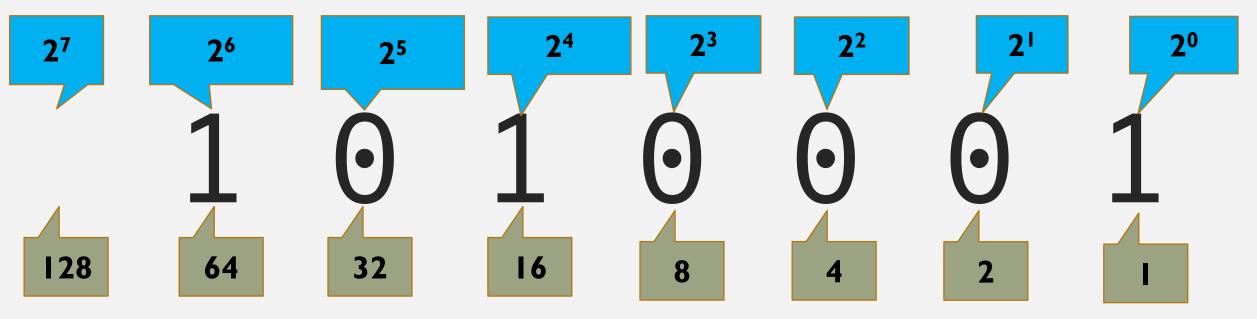
28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	I



## Converting Binary to Decimal

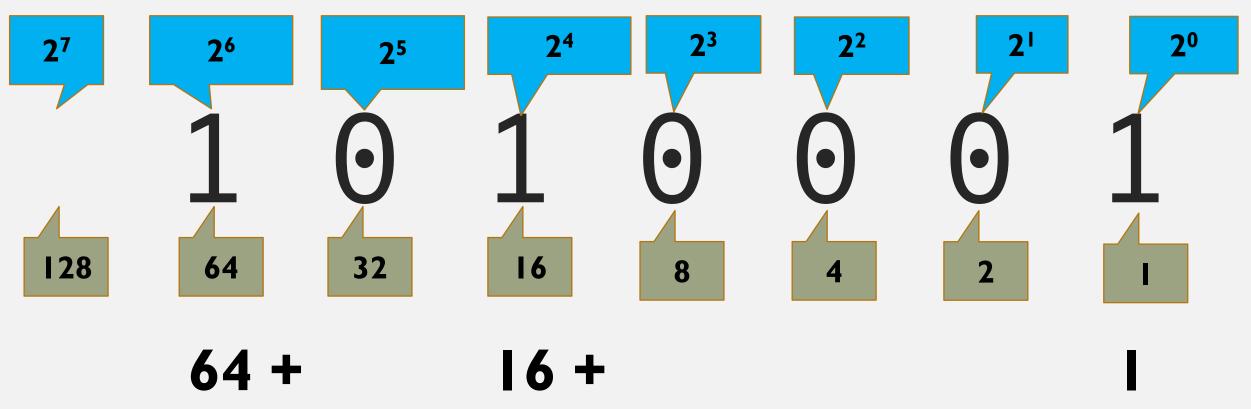
## Here's our binary for 81 - is it correct?

Binary uses base2 numbers (start from the right)



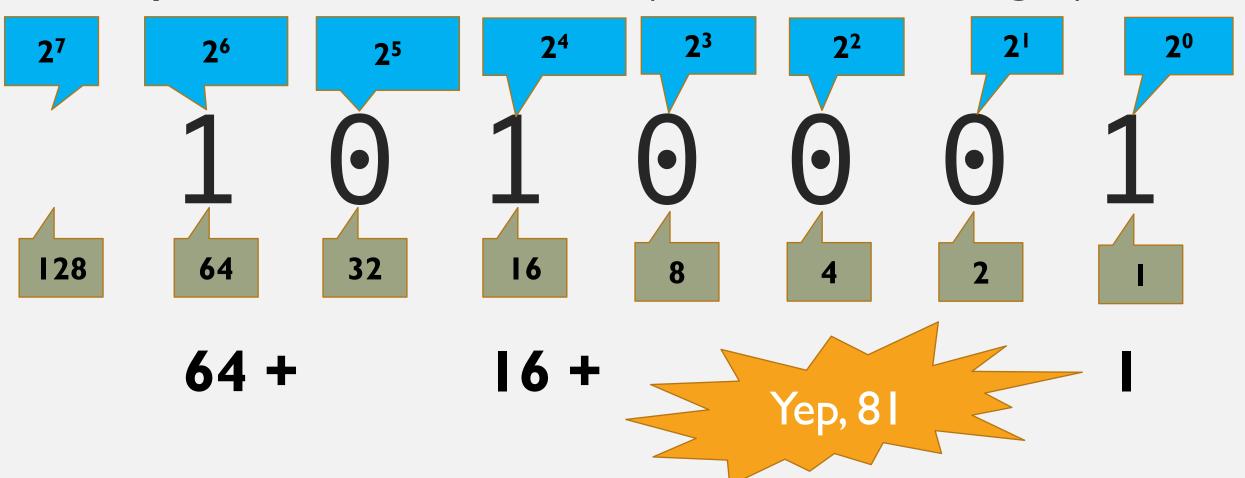
## Here's our binary for 81. Is it correct?

Binary uses base2 numbers (start from the right)



## Here's our binary for 81. Is it correct?

Binary uses base2 numbers (start from the right)



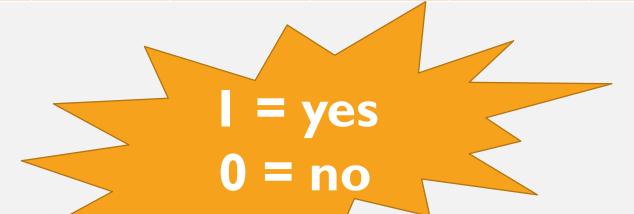
• This table is cleaner, but the exact same idea

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	l
			0	I	0	0	0	I

Binary 81

• This table is cleaner, but the exact same idea

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	2 <sup>0</sup>
256	128	64	32	16	8	4	2	I
		1	0	1	0	0	0	I
0	0	64	0	16	0	0	0	1



• This table is cleaner, but the exact same idea

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	I
		1	0	1	0	0	0	I
0	0	64	0	16	0	0	0	1
81								

## This table makes it easy

- Any binary number up to 8 bits
  - And you could make this table as many columns as necessary

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	I
I	I	0	0	I	I	I	0	0

## This table makes it easy

- Any binary number up to 8 bits
  - And you could make this table as many columns as necessary

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	I
1	1	0	0	1	I	1	0	0
256	128	0	0	16	8	4	0	0

## This table makes it easy

- Any binary number up to 8 bits
  - And you could make this table as many columns as necessary

28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
256	128	64	32	16	8	4	2	1
1	1	0	0	1	1	1	0	0
256	128	0	0	16	8	4	0	0

# Converting Hex to Decimal

#### Hex to Decimal

Base<sub>16</sub> to Base<sub>10</sub>

- Three to check!
  - Is 81 really 0x51?
  - Is 187 really 0xBB?
  - Is 58,345 really 0xE3E9?

Decimal	Hex
0	0
1	I
2	2
3	3
4	4
5	5
6	6
7	7

Decimal	Hex
8	8
9	9
10	Α
11	В
12	С
13	D
14	Е
15	F

- Hex is base<sub>16</sub>
- · It's still a positional system; just like binary and decimal

• When you see 0xE3E9, you take it position by position

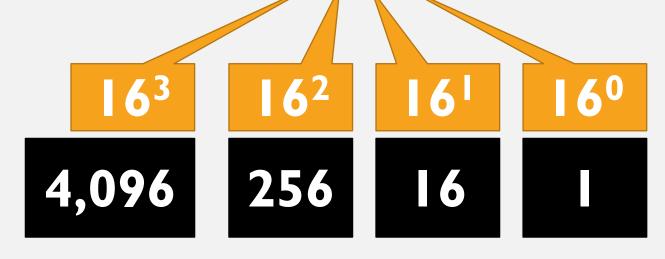
- Hex is base<sub>16</sub>
- · It's still a positional system; just like binary and decimal

• When you see 0xE3E9, you take it position by position



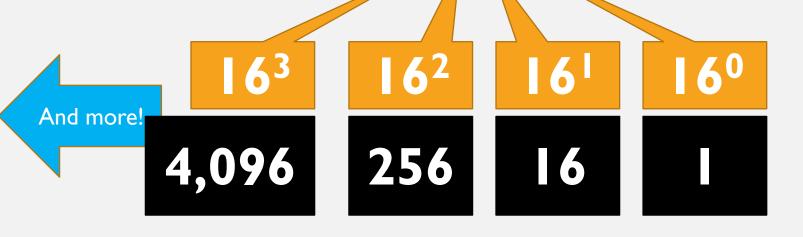
- Hex is base<sub>16</sub>
- · It's still a positional system; just like binary and decimal

When you see 0xE3E9, you take it position by position

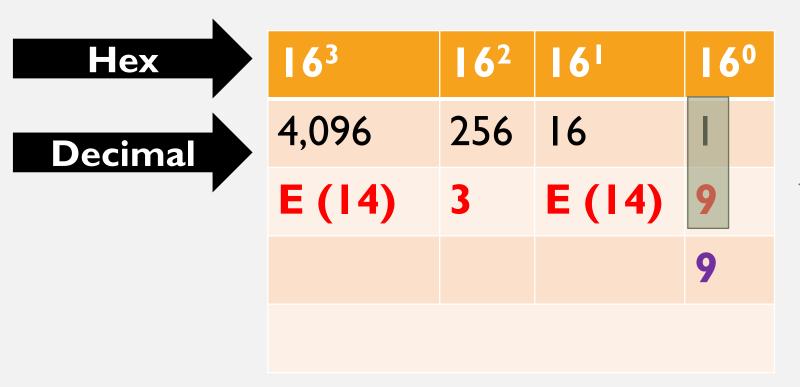


- Hex is base<sub>16</sub>
- · It's still a positional system; just like binary and decimal

• When you see 0xE3E9, take it position by position



Hex	. •		164				
Decimal	16,777,216	1,048,576	65,536	4,096	256	16	I
•					3		





Hex	163	16 <sup>2</sup>	161	16 <sup>0</sup>
Decimal	4,096	256	16	I
	E (14)	3	E (14)	9
			224	9

Hex	163	16 <sup>2</sup>	161	160
Decimal	4,096	256	16	I
	E (14)	3	E (14)	9
		768	224	9

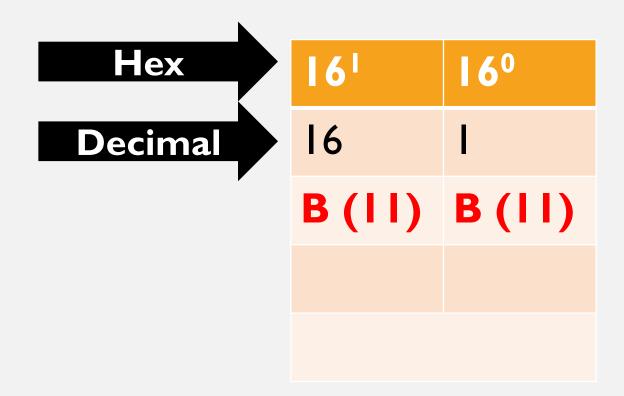
Hex	163	16 <sup>2</sup>	161	160
Decimal	4,096	256	16	I
	E (14)	3	E (14)	9
	57,344	<b>768</b>	224	9

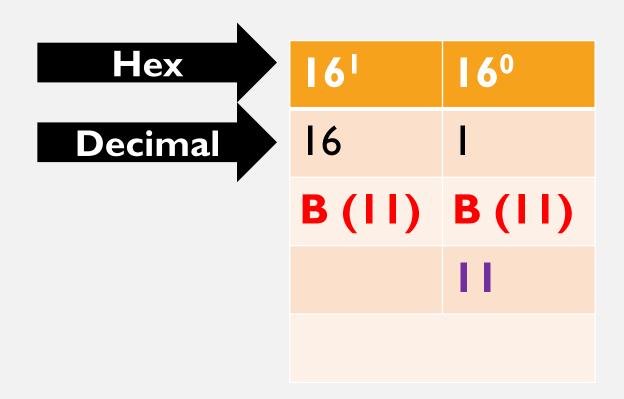
Hex	163	16 <sup>2</sup>	161	160	
Decimal	4,096	256	16	1	
	E (14)	3	E (14)	9	
	57,344	<b>768</b>	224	9	
	58,345				

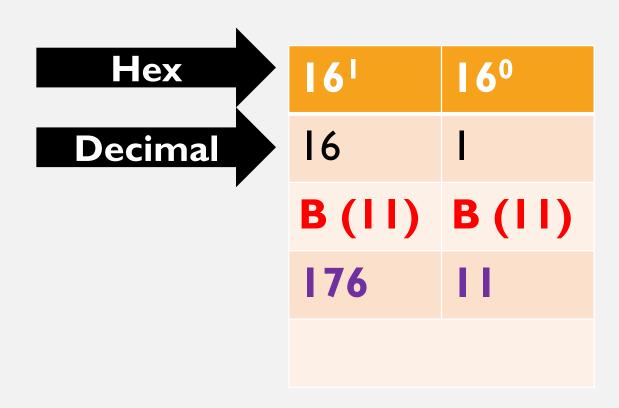


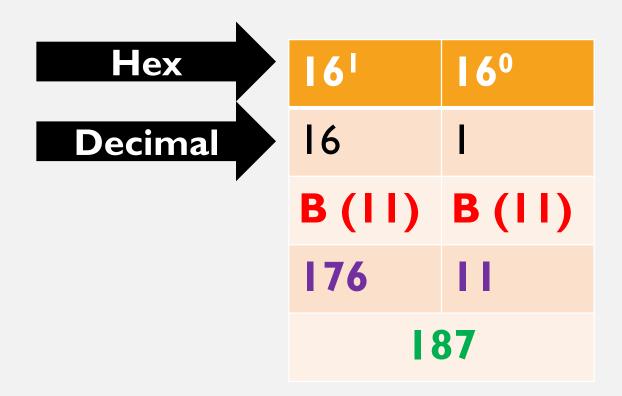
Hex	
Decimal	
Decimal	

•	166	165	164	163	162	161	160
•	16,777,216	1,048,576	65,536	4,096	256	16	I
						В	В









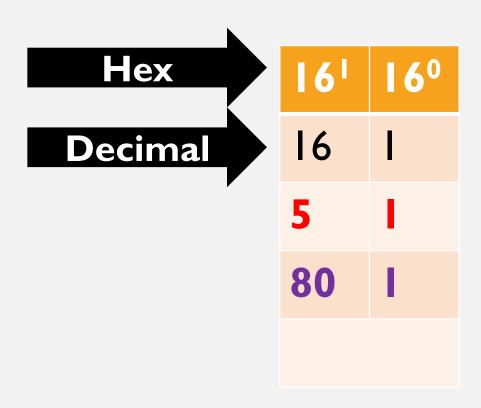
Consider 0x51 (we think it's 81)

Hex	
Decimal	

•	166	165	164	163	16 <sup>2</sup>	161	16º
<b>&gt;</b>	16,777,216	1,048,576	65,536	4,096	256	16	I
						5	I

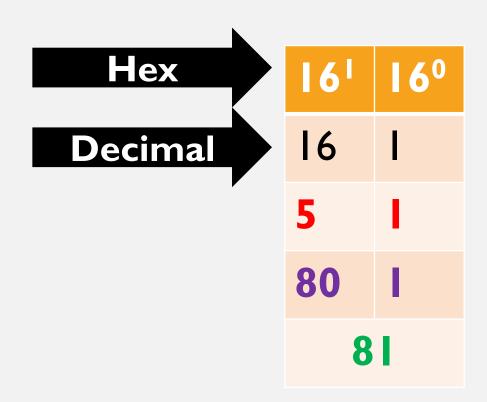
#### Let's look at that same idea in table format

• Consider 0x51 (we think it's 81)



#### Let's look at that same idea in table format

• Consider 0x51 (we think it's 81)



#### So far

- Conversions:
  - Decimal to Binary
  - Binary to Decimal
  - Decimal to Hex
  - Hex to Decimal
- Left to do:
  - Hex to Binary
  - Binary to Hex

# Converting Hex to Binary

Base<sub>16</sub> to Base<sub>2</sub>

- Think of the character sets of each
  - Hex has 16 characters (0-9, A-F)
  - Binary has 2 characters (0 and 1)

• How many combinations of 0 and 1 are there in a 2-bit number?

• How many combinations of 0 and 1 are there in a 2-bit number?

00, 01, 10, 11

• How many combinations of 0 and 1 are there in a 2-bit number?

00, 01, 10, 11



• How many combinations of 0 and 1 are there in a 2-bit number?

```
00, 01, 10, 11
```

0, 1, 2, 3



• How many combinations of 0 and 1 are there in a 2-bit number?

00, 01, 10, 11

• How about a 3-bit number?

• How many combinations of 0 and 1 are there in a 2-bit number?

00, 01, 10, 11

How about a 3-bit number?
000, 001, 010, 011, 100, 101, 110, 111

• See a pattern yet? ©

• How many combinations of 0 and 1 are there in a 2-bit number?

```
00, 01, 10, 11
```

• How about a 3-bit number?

```
000, 001, 010, 011, 100, 101, 110, 111
0, 1, 2, 3, 4, 5, 6, 7
```

• See a pattern yet? ©

• How about a 4-bit number?

How about a 4-bit number?0000, 0001, 0010, 0011,

0100, 0101, 0110, 0111,

1000, 1001, 1010, 1011,

1100, 1101, 1110, 1111

• How about a 4-bit number?

```
0000, 0001, 0010, 0011, → 0, 1, 2, 3
0100, 0101, 0110, 0111, → 4, 5, 6, 7
1000, 1001, 1010, 1011, → 8, 9, 10, 11
1100, 1101, 1110, 1111 → 12, 13, 14, 15
```

• How about a 4-bit number?

```
0000, 0001, 0010, 0011, → 0, 1, 2, 3
0100, 0101, 0110, 0111, → 4, 5, 6, 7
1000, 1001, 1010, 1011, → 8, 9, 10, 11
1100, 1101, 1110, 1111 → 12, 13, 14, 15
```

Look familiar?

• How about a 4-bit number?

```
0000, 0001, 0010, 0011, → 0, 1, 2, 3
0100, 0101, 0110, 0111, → 4, 5, 6, 7
1000, 1001, 1010, 1011, → 8, 9, A, B
1100, 1101, 1110, 1111 → C, D, E, F
```

Hex!

## Hex to Binary -- OK with this?

Binary	Hex
0000	0
0001	I
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binary	Hex
1000	8
1001	9
1010	Α
1011	В
1100	С
1101	D
1110	E
Ш	F

## Hex to Binary -- OK with this?

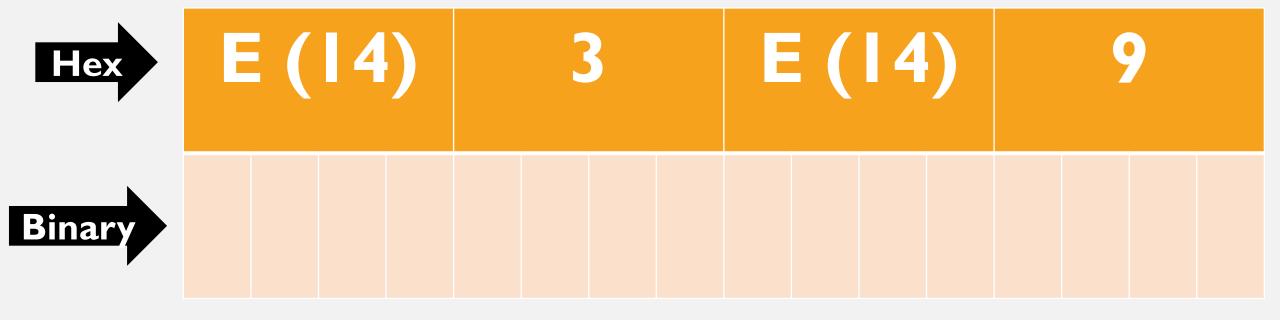
Dec	Binary	Hex
0	0000	0
I	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7

# Each hex digit is 4-bits!

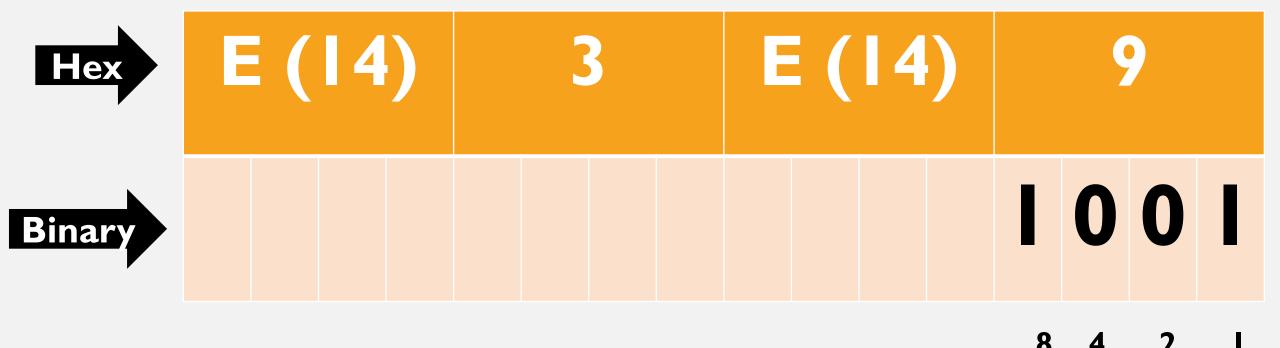
Dec	Binary	Hex
8	1000	8
9	1001	9
10	1010	Α
11	1011	В
12	1100	С
13	1101	D
14	1110	Е
15	1111	F

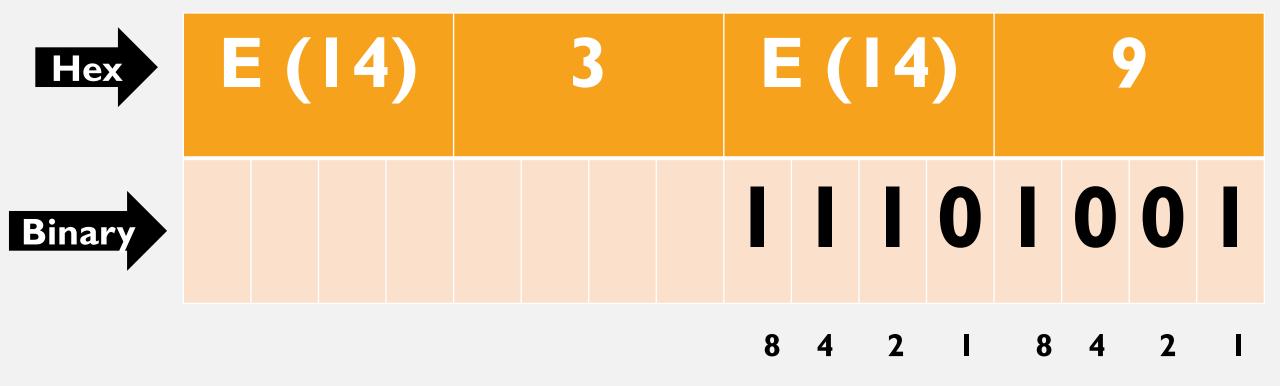
Hex	E (14)	3	E (14)	9	
Binary					

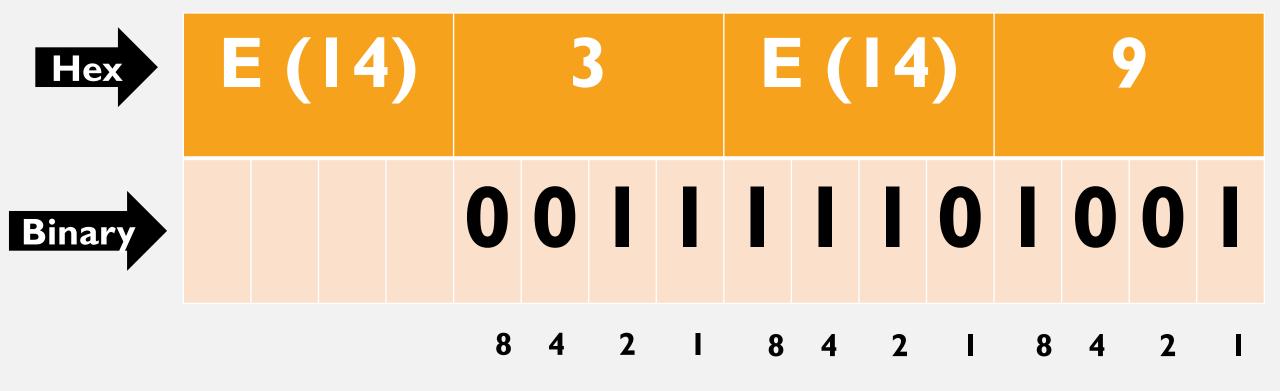
Consider 0xE3E9

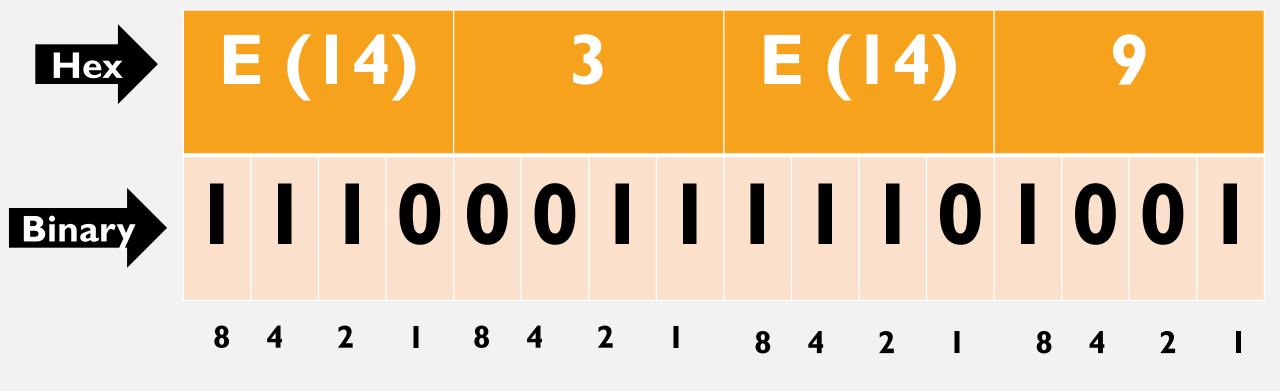


3 4 2 I









$$0xE3E9 = 11100011111101001$$

This is 2 to what power?

$$0xE3E9 = 11100011111101001$$

$$0xE3E9 = 11100011111101001$$

$$2^{15} = 32,768$$

$$2^{14} = 16,384$$

$$0xE3E9 = 1110001111101001$$



 $2^{14} = 16,384$ 

# 0xE3E9 = 11100011111101001

- Hex is much more compact compared to binary!
- How about decimal?

$$0xE3E9 = 11100011111101001$$

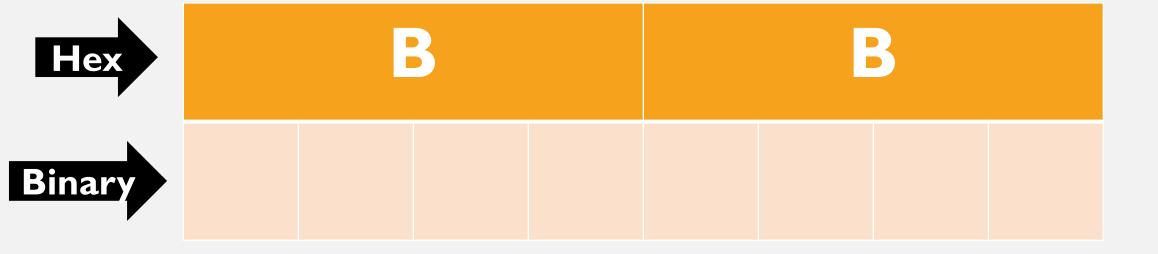
- Hex is much more compact compared to binary!
- How about decimal?

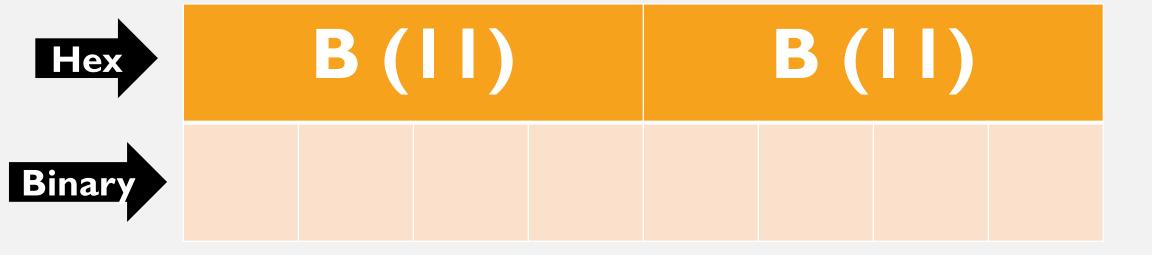
$$0xE3E9 = 11100011111101001 = 58,345$$

Hex

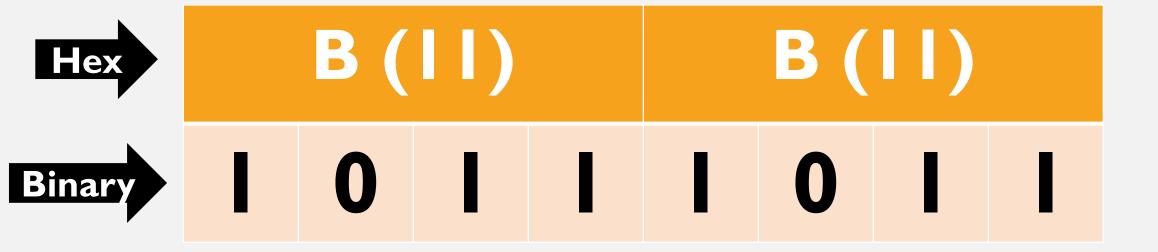
Binary

Decimal





Hex	B (II)			B (II)			
Binary					0		Ī



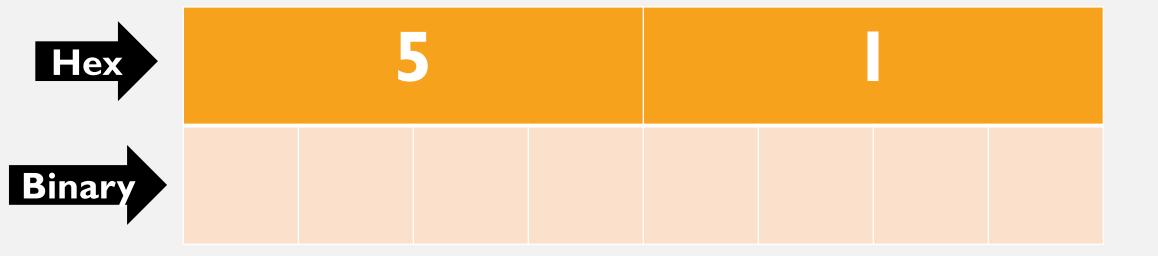
Hex	B (11)				<b>B</b> (	11)		
Binary						0		
Decimal	128	64	32	16	8	4	2	I

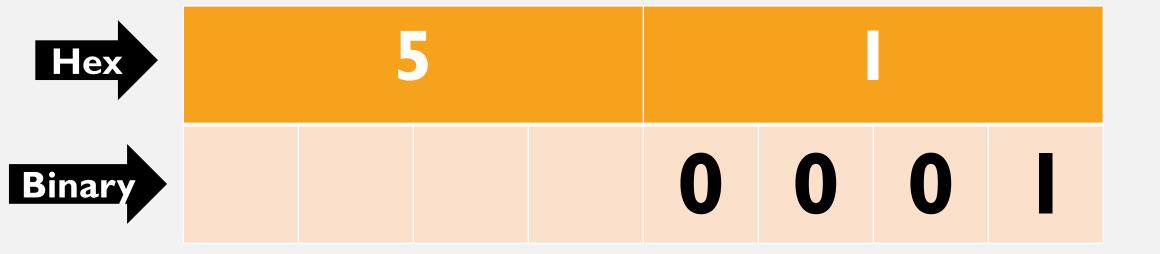
Consider 0xBB

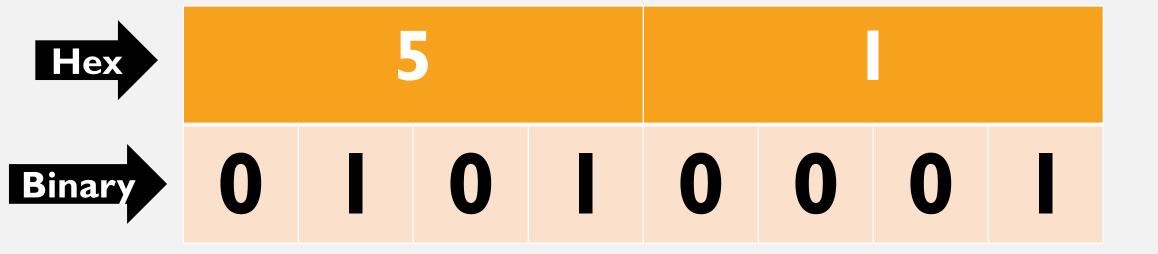
Hex		<b>B</b> (				<b>B</b> (		
Binary						0		
Decima	128	64	32	16	8	4	2	I

128+32+16+8+2+1=187

Consider 0x51







Hex	5							
Binary	0		0		0	0	0	
Decimal	128	64	32	16	8	4	2	Í

Hex	5							
Binary	0		0		0	0	0	
Decimal	128	64	32	16	8	4	2	I
64 + 16 + 1 = 81								

Base<sub>2</sub> to Base<sub>16</sub>

Remember the character sets?

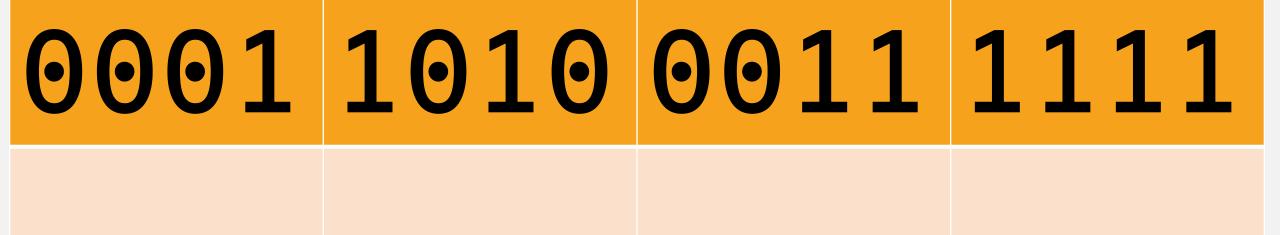
 Longest representation to the shortest

Binary	Hex
0000	0
0001	I
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binary	Hex
1000	8
1001	9
1010	A
1011	В
1100	С
1101	D
1110	Е
1111	F

#### 0001 1010 0011 1111 to 0x?

- I. How many bits are in the binary number?
  - Every 4 bits is one hex digit
- 2. Calculate each hex digit based on binary position
  - 0000...IIII is converted to 0...F



- I. How many bits are in the binary number?
  - Every 4 bits is one hex digit
- 2. Calculate each hex digit based on binary position
  - 0000...IIII is converted to 0...F



- I. How many bits are in the binary number?
  - Every 4 bits is one hex digit (4 bits in the number above!)
- 2. Calculate each hex digit based on binary position
  - 0000...IIII is converted to 0...F



- I. How many bits are in the binary number?
  - Every 4 bits is one hex digit (4 bits in the number above!)
- 2. Calculate each hex digit based on binary position
  - 0000... | | | | is converted to 0... F (Start from the right)

0001	1010	0011	1111
			8+4+2+1

0001	1010	0011	1111
			8+4+2+1
			15(F)

0001	1010	0011	1111
		0+0+2+1	8+4+2+1
			15(F)

0001	1010	0011	1111
		0+0+2+1	8+4+2+1
		3	15(F)

0001	1010	0011	1111
	8+0+2+0	0+0+2+1	8+4+2+1
		3	15(F)

0001	1010	0011	1111
	8+0+2+0	0+0+2+1	8+4+2+1
	10(A)	3	15(F)

0001	1010	0011	1111
0+0+0+1	8+0+2+0	0+0+2+1	8+4+2+1
	10(A)	3	15(F)

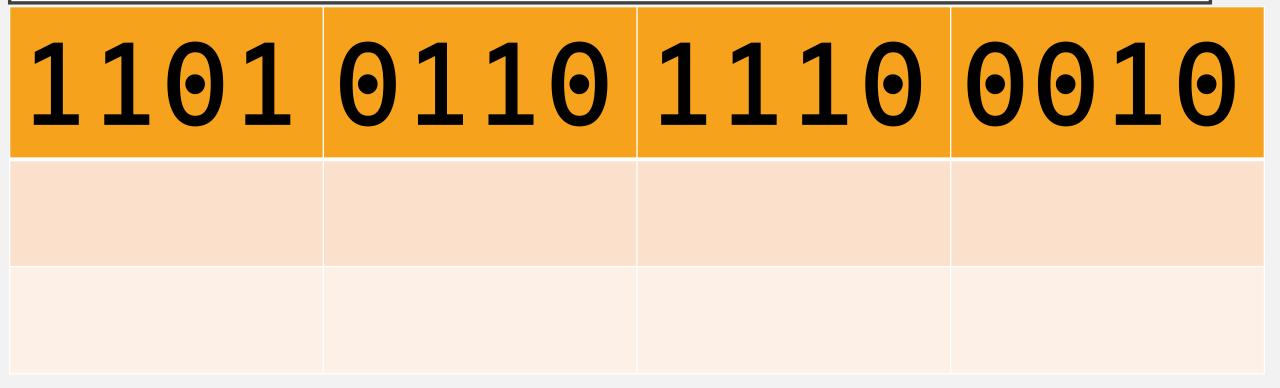
0001	1010	0011	1111
0+0+0+1	8+0+2+0	0+0+2+1	8+4+2+1
1	10(A)	3	15(F)

0001	1010	0011	1111
0+0+0+1	8+0+2+0	0+0+2+1	8+4+2+1
1	10(A)	3	15(F)

# 0x1A3F

#### And another one!

•One more time to make sure we know what we're doing with this conversion! ©



1101	0110	1110	0010
			0+0+2+0

1101	0110	1110	0010
			0+0+2+0
			2

1101	0110	1110	0010
		8+4+2+0	0+0+2+0
			2

1101	0110	1110	0010
		8+4+2+0	0+0+2+0
		14(E)	2

1101	0110	1110	0010
	0+4+2+0	8+4+2+0	0+0+2+0
		14(E)	2

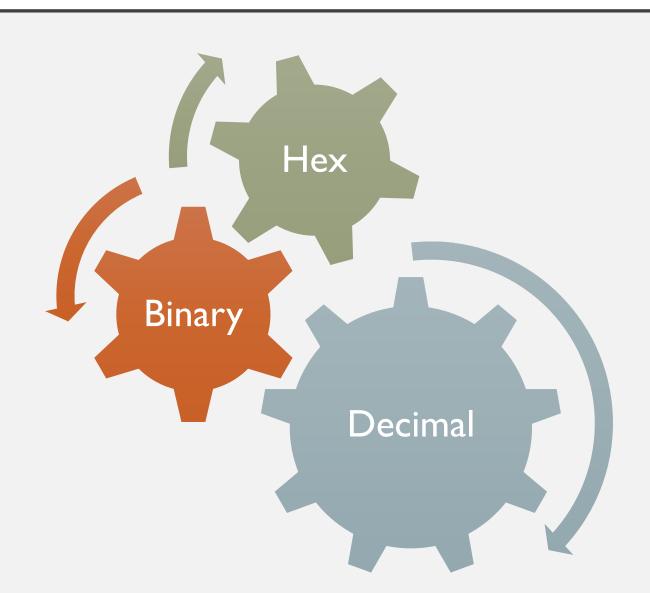
1101	0110	1110	0010
	0+4+2+0	8+4+2+0	0+0+2+0
	6	14(E)	2

1101	0110	1110	0010
8+4+0+1	0+4+2+0	8+4+2+0	0+0+2+0
	6	14(E)	2

1101	0110	1110	0010
8+4+0+1	0+4+2+0	8+4+2+0	0+0+2+0
13(D)	6	14(E)	2

# 0xD6E2

#### That's all conversions we're going to cover



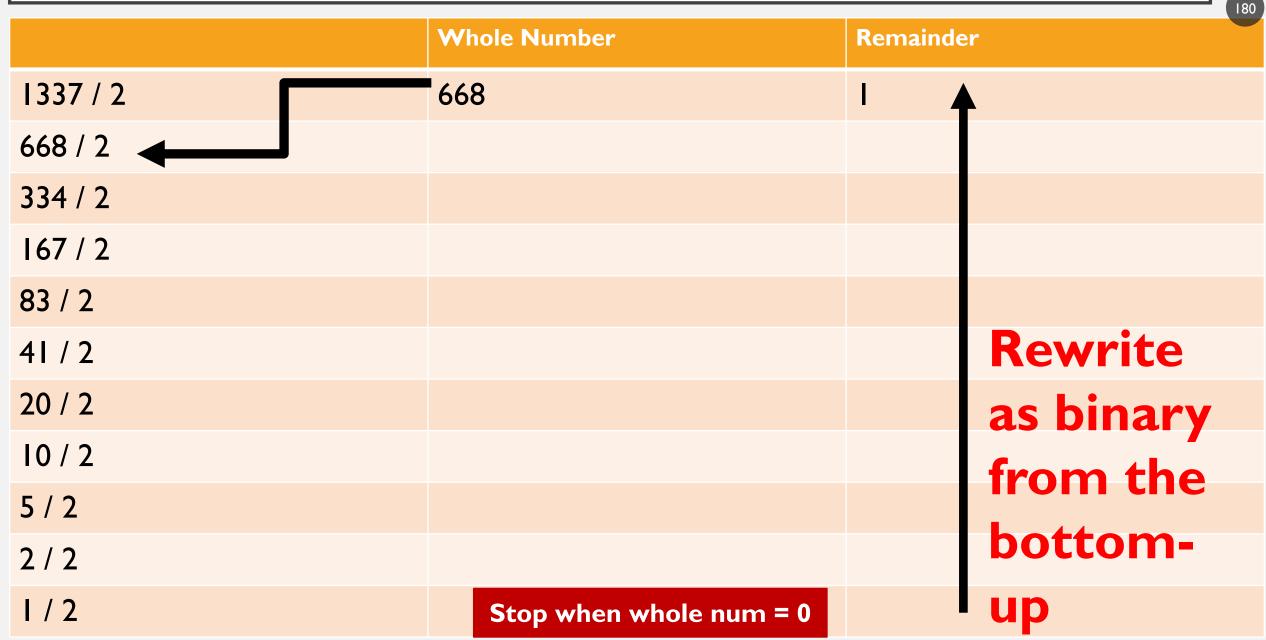
## Was that great fun or what!?

Challenges

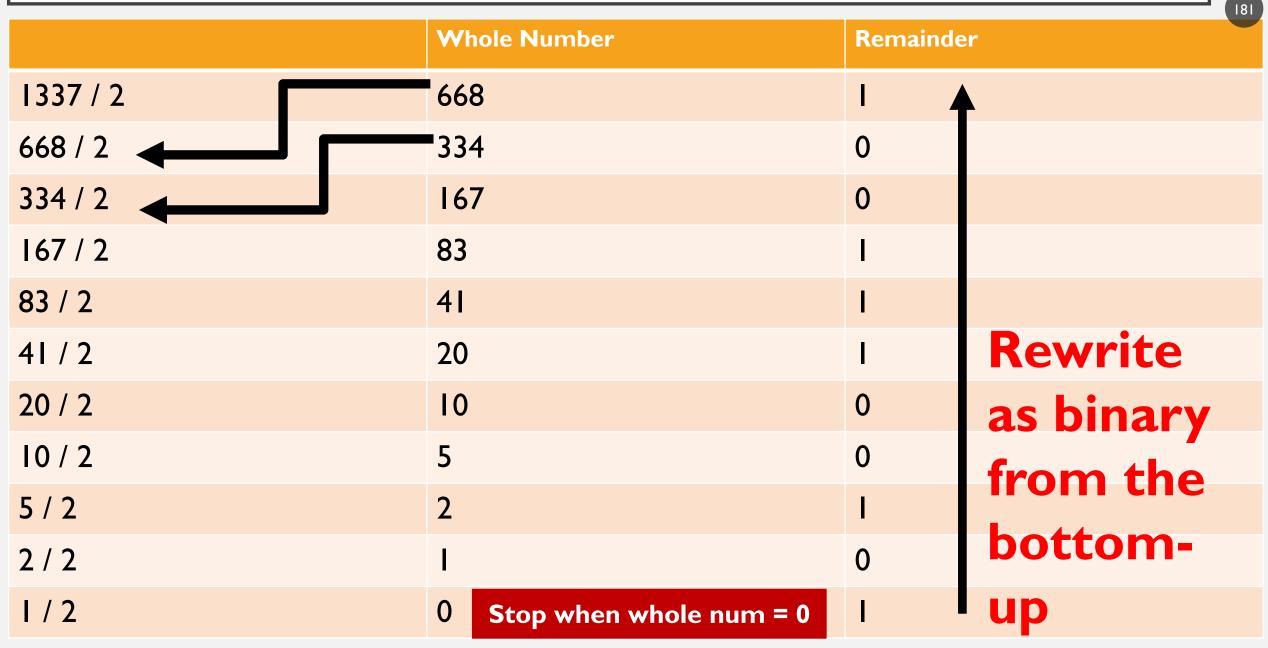
#### Converting Decimal to Binary

num I: What is the 6th whole number when you convert I337 from decimal to binary?

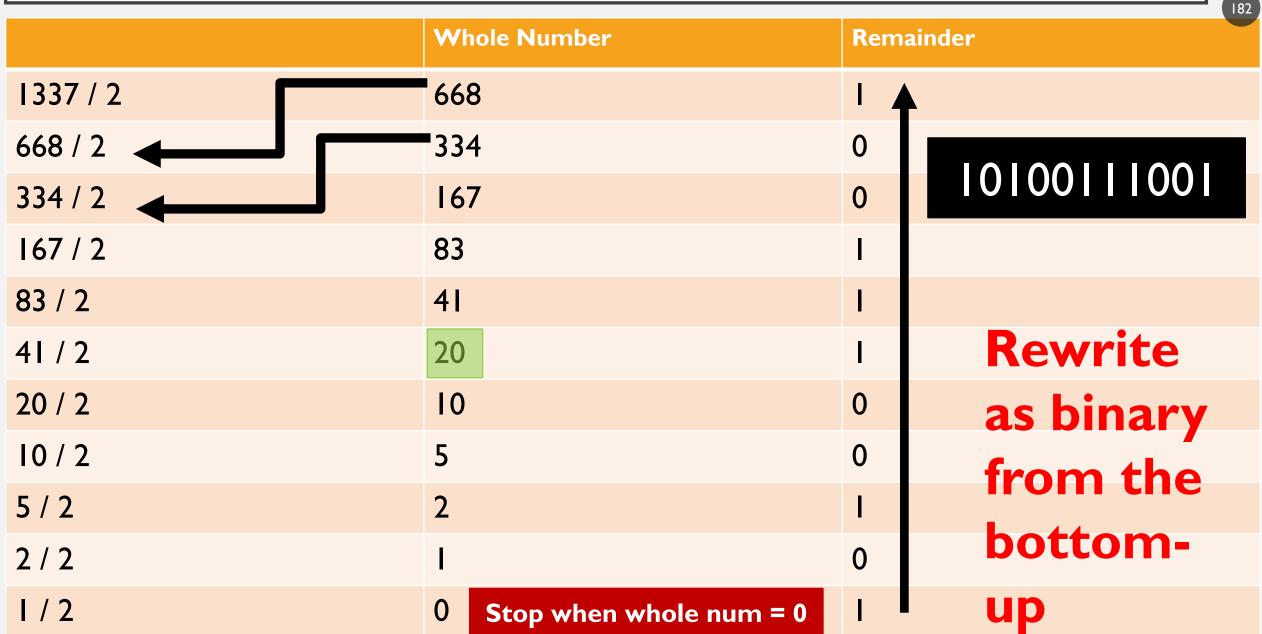
#### Converting Decimal to Binary: Repeated Divide by 2



## Converting Decimal to Binary: Repeated Divide by 2



## Converting Decimal to Binary: Repeated Divide by 2



## Converting Binary to Decimal

num2:What is the value of the 2<sup>11</sup> digit when you convert 1101110011100 from binary to decimal?

# Converting Binary to Decimal: This table makes it easy

212	211	210	29	<b>2</b> <sup>8</sup>	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
4096	2048	1024	512	256	128	64	32	16	8	4	2	I

## Converting Binary to Decimal: This table makes it easy

212	211	210	29	28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
4096	2048	1024	512	256	128	64	32	16	8	4	2	I
1	I	0	1	1	1	0	0	I	1	1	0	0
4096	2048	0	512	256	128	0	0	16	8	4	0	0
				7,068								

# Converting Binary to Decimal: This table makes it easy

212	211	210	29	28	27	26	<b>2</b> <sup>5</sup>	24	<b>2</b> <sup>3</sup>	<b>2</b> <sup>2</sup>	21	<b>2</b> <sup>0</sup>
4096	2048	1024	512	256	128	64	32	16	8	4	2	I
I	I	0	I	I	I	0	0	I	1	1	0	0
4096	2048	0	512	256	128	0	0	16	8	4	0	0
				7,068								

## Converting Decimal to Hex

num3: What is the 3rd whole number when you convert the 2010 census population of Baltimore, MD from decimal to hex?

# Converting Decimal to Hex: Repeated Divide by 16

	Whole Number	Remainder
620,961 / 16	38,810	
38,810 / 16		
	Stop when whole num = 0	

# Converting Decimal to Hex: Repeated Divide by 16

	Whole Number	Remainder
620,961 / 16	38,810	
38,810 / 16	2,425	10 (A)
2,425 / 16	151	9
151/16	9	7
9/16	O Stop when whole num = 0	9

# Converting Decimal to Hex: Repeated Divide by 16

	Whole Number	Remainder
620,961 / 16	38,810	1
38,810 / 16	2,425	10 (A)
2,425 / 16	151	9
151/16	9	7
9/16	O Stop when whole num = 0	9 0x979AI

## Converting Hex to Decimal

num4: What is the decimal value of the 16<sup>3</sup> digit when you convert 0x81BEEF from hex to decimal?

Consider 0x81BEEF

Hex Dec

165	164	163	16 <sup>2</sup>	161	160

### Consider 0x81BEEF

Hex	
Dec	

165	164	163	16 <sup>2</sup>	161	160
1,048,576	65,536	4,096	256	16	I
8	1	<b>B</b> (II)	E (14)	E (14)	F (15)

### Consider 0x81BEEF

Hex	
Dec	
	7

165	164	163	162	161	16 <sup>0</sup>		
1,048,576	65,536	4,096	256	16	I		
8	1	<b>B</b> (II)	E (14)	E (14)	F (15)		
8,388,608	65,536	45,056	3,584	224	15		
8,503,023							

### Consider 0x81BEEF

Hex	
	7
_	
Dec	

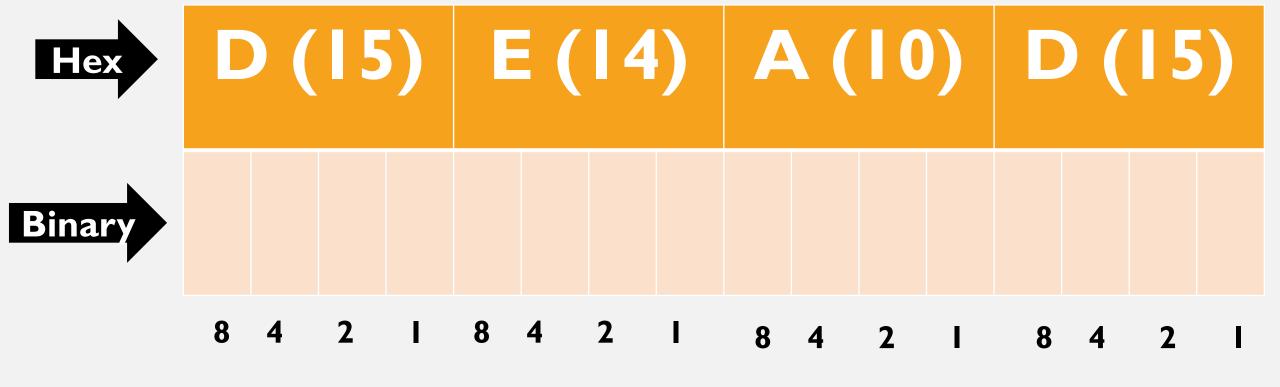
165	164	163	16 <sup>2</sup>	161	160	
1,048,576	65,536	4,096	256	16	I	
8	I	Ш	14	14	15	
8,388,608	65,536	45,056	3,584	224	15	
8,503,023						

## Converting Hex to Binary

How many 2's are included in the 4-bit numbers during conversion of 0xDEAD to binary?

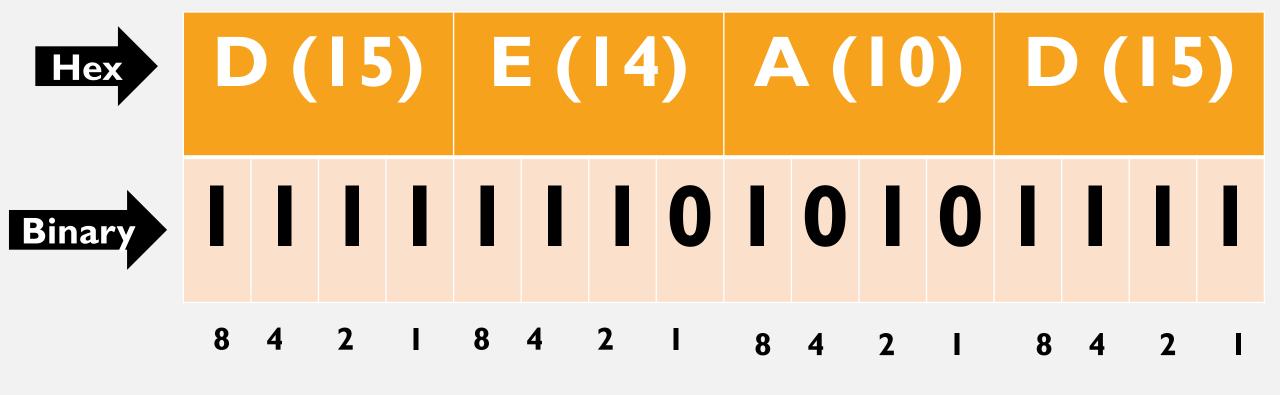
## Converting Hex to Binary: Here it is!

Consider 0xDEAD



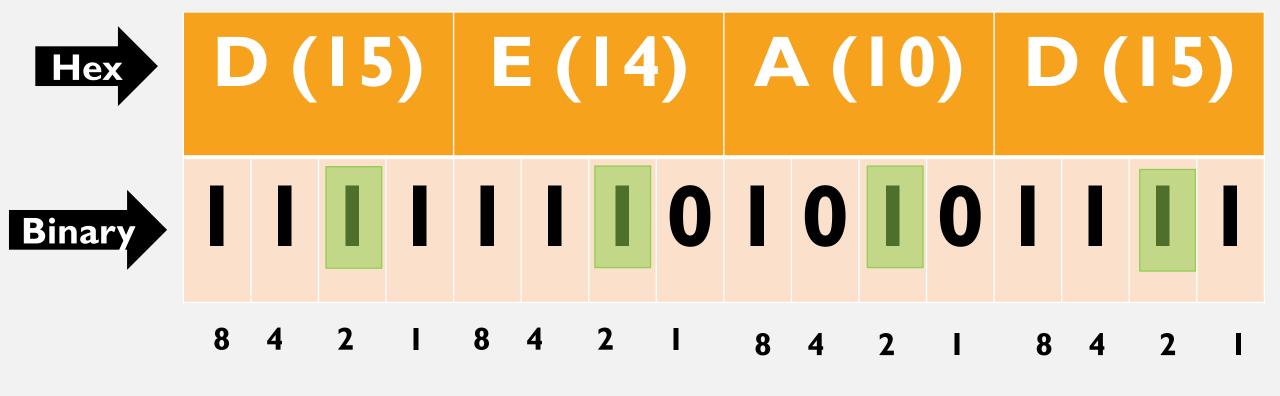
## Converting Hex to Binary: Here it is!

Consider 0xDEAD



## Converting Hex to Binary: Here it is!

Consider 0xDEAD



## Converting Binary to Hex

What Michael Jackson song released on my 8<sup>th</sup> birthday (August 31, 1987) is 101110101101 in binary?

## Converting Binary to Hex

1011	1010	1101
8+4+2+1	8+0+2+0	8+4+0+1

## Converting Binary to Hex

# **OXBAD**

## Friday Fun Day

Team C coding for candy and prizes

## As we get settled in today....

- 1. Log into ialab.dsu.edu
- 2. Log into Learn
- 3. Log into your Windows VM
- 4. Open Visual Studio Code (shortcut on Desktop)
- 5. Open x86 Native Tools Command Prompt (pinned to Start Menu)

## The Set Up (You Need This)

- 1. Draw card (A-4 in all suits)
- 2. Find partner (value & color)
- 3. Pick location (any table)
- 4. Pick problem (keep it secret)
- 5. Add one of you as a github collaborator on my laptop

## Why are we doing this?

- 1. Outcomes that you don't understand (potentially)
- 2. Exceptionally vague instructions (potentially)
- 3. Sitting at a table you don't like (potentially)
- 4. Work you don't like (potentially)
- 5. Languages you don't like (potentially)
- 6. Team with people you don't like (potentially)
- 7. You don't like to speak (potentially)
- \*\*We're only doing this for less than one day
- \*\*I offer candy and prizes

### Round I

- 1. Write a simple C program in VS Code that solves your math problem
- Use the number set conversions we covered earlier to add a little spice
- 3. Save r1-color-value.c (r1-red-4)
- 4. Compile with cl in cmd prompt
- 5. Share your .exe as a release at github.com/joshpauli/cae23
- 4. Tag it r1
- 5. Title it r1-color-value

```
208
```

```
#include <stdio.h>
int main()
   int x = 4;
   printf("The answer is %d", x);
   return 0;
```

cl is the installed compiler

### Round I

- 1. Download & reverse the same value as your team r1-black-4 → r1-red-4
- 2. Once you have it, go check with them; many lulz will be had
- 3. Grab a prize
- 4. Plug in and talk us through it!
- 5. Solve more r1 .exe or help a team

## Round 2

- 1. Harder problems
- 2. Obfuscate code techniques
- 3. Compiler options
- 4. Release to github as r2
- 5. RE your same color neighbor
- r2-black-A → r2-black-2
- r2-black-3 → r2-black-4
- $r2-red-A \rightarrow r2-red-2$
- r2-red-3 → r2-red-4

## cl Compiler Options

- /MD Causes the application to use the multithread-specific and DLL-specific version of the run-time library. Applications compiled with this option are statically linked to MSVCRT.lib. This library provides a layer of code that enables the linker to resolve external references.
- /Od Turns off all optimizations in the program and speeds compilation.
- /Zi The /Zi option produces a separate PDB file that contains all the symbolic debugging information for use with the debugger. The debugging information isn't included in the object files or executable, which makes them much smaller.
- /FA Creates a listing file.
- /Fa Sets the listing file name.
- /Fe Renames the executable file.
- /Fo Creates an object file.