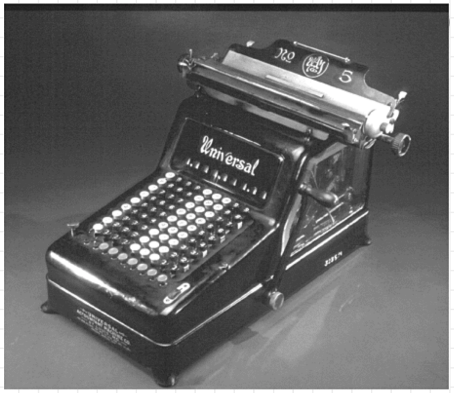


Adding Machine 1880-1920 computers.org

CSCI 2824: Discrete Structures

Lecture 2: Binary Numbers. Intro Python

Hello world!

My name is IOANA FLEMING

You can find me at:

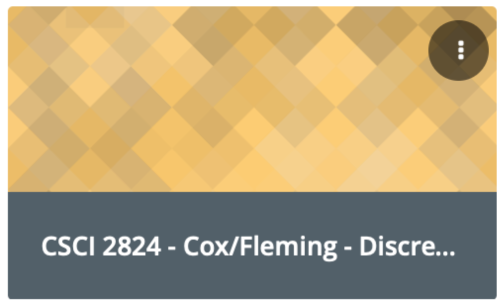
ioana.fleming@colorado.edu

ECOT 735

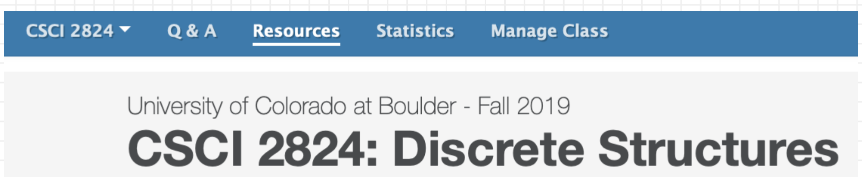
Course Logistics - Platforms

1) Moodle – Online Homework, Online Quizlets, Grades

CSCI2824-Fall2019



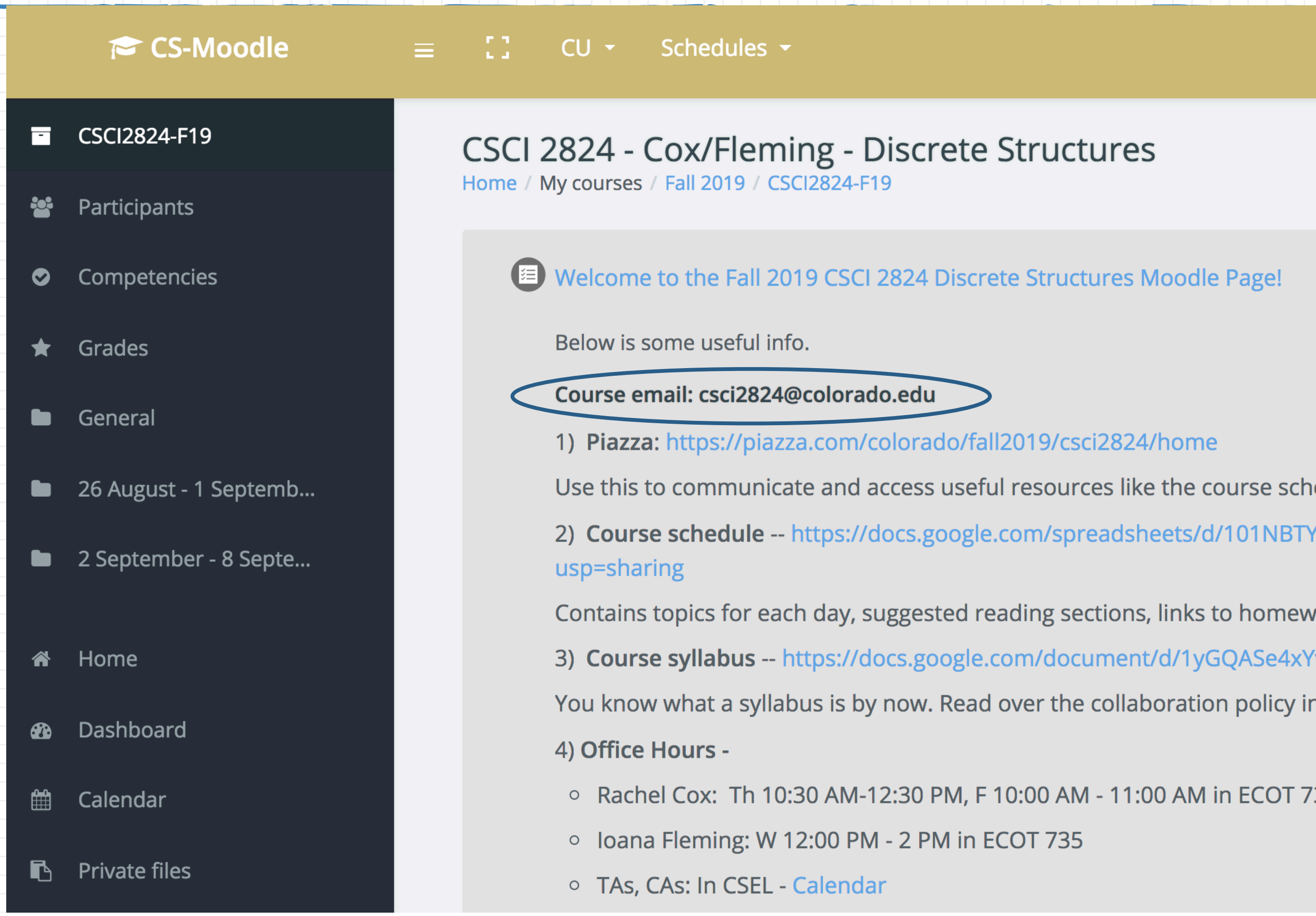
2) Piazza– Class discussion forum



3) Gradescope – Submission of written homework

gradescope

Course email



2824 Workgroup

- 100 minutes
- Th @ 5pm or F @ 1pm
- Solve problems with uTA
- Pass/Fail
- Cannot miss more than 2/15 sessions

Last time

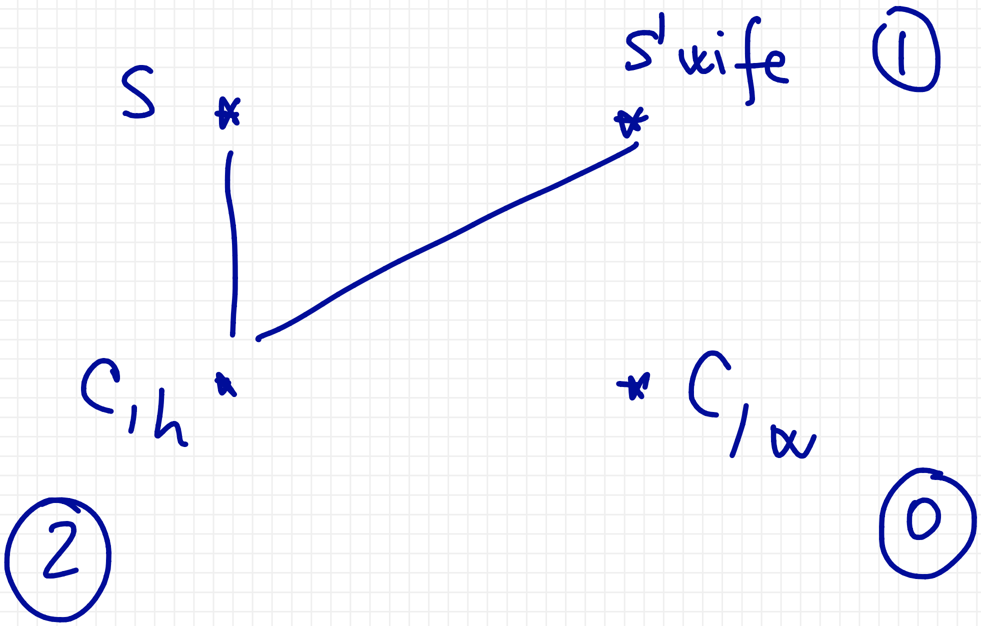
Mr. Smith and his wife invited four other couples for a party. When everyone arrived, some of the people in the room shook hands with some of the others. Of course, nobody shook hands with their spouse and nobody shook hands with the same person twice.

After that, Mr. Smith asked everyone how many times they shook someone's hand. He received different answers from everybody .

How many times did Mrs. Smith shake someone's hand?

4

Last time: How many times did Mrs. Smith shake someone's hand?



0, 1, 2

Binary Representation of Numbers

An Algorithm for Converting Decimal Integers to Binary

Let N be a nonnegative integer. Move from right to left.

Is N even? or odd?

> If N is even, set bit to 0, reset $N = \frac{N}{2}$
>
> If N is odd, set bit to 1, reset $N = \frac{N-1}{2}$

Move left to the next bit

Repeat until N = 0

2^0

Fill bits in this direction

$$1*2^0 + 1*2^1 + 1*2^3 + 1*2^5 + 1*2^6 + 1*2^7 =$$

$$= 1 + 2 + 8 + 32 + 64 + 128 = 235 \checkmark$$

Example:

Convert 235 from decimal to binary using the algorithm we just defined

	Q	R
$\frac{235}{2}$	$= 117$	1
$\frac{117}{2}$	$= 58$	1
$\frac{58}{2}$	$= 29$	0
$\frac{29}{2}$	$= 14$	1
$\frac{14}{2}$	$= 7$	0
$7/2$	$= 3$	1
$\frac{3}{2}$	$= 1$	1
$\frac{1}{2}$	$= 0$	1

1	1	1	0	1	0	1	1	.			
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}

Example:

Convert 1100101 from binary to decimal

$$1 * 2^0 + 1 * 2^2 + 1 * 2^4 + 1 * 2^5 + 1 * 2^6 =$$

$$= 1 + 4 + 16 + 32 + 64 =$$

$$= 117$$

$$110101_2 = 117_{10}$$

	1	1	0	0	1	0	1	.			
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}

1^{st} bit is reserved for the sign

$+ : 0$

$- : 1$

Example:

What's the largest number you can store as a 32-bit signed int?

2 bits $11_2 \longrightarrow 3_{10} = 4-1$

3 bits $111_2 \longrightarrow 7_{10} = 8-1$

4 bits $1111_2 \longrightarrow 15_{10} = 16-1$

...

n bits $\longrightarrow = 2^n - 1$

$$\boxed{2^{31} - 1}$$

Example:

What's the largest number you can store as a 32-bit signed int?

The number 2,147,483,647 is the eighth *Mersenne prime*, equal to $2^{31} - 1$.

Marin Mersenne

Born	8 September 1588 Oizé, Maine
Died	1 September 1648 (aged 59) Paris
Nationality	French
Known for	Acoustics, Mersenne primes

Fractions in binary

$0.75_{10} = 0.7 + 0.05 =$

$= 7 * 10^{-1} + 5 * 10^{-2}$

$0.75 - 0.5 = 0.25$

$0.25 - 0.25 = 0$

$(0.11)_2 \longrightarrow 1*2^{-1} + 1*2^{-2} = 0.5 + 0.25 =$

$= (0.75)_{10}$

0	0	0						.	1	1	0	0	0	0	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}

$\longleftarrow$ Integer

$\longrightarrow$ Fractional part : < 1

Powers of 2:

$2^{-1} = \frac{1}{2} = 0.5$

$2^{-2} = \frac{1}{4} = 0.25$

$2^{-3} = \frac{1}{8} = 0.125$

$2^{-4} = \frac{1}{16}$

$2^{-5} = \frac{1}{32}$

$2^{-6} = \frac{1}{64}$

$2^{-7} = \frac{1}{128}$

$\vdots$

Example

Convert 0.75 from decimal to binary

Powers of 2:

$$2^{-1} = \frac{1}{2}$$
$$2^{-2} = \frac{1}{4}$$
$$2^{-3} = \frac{1}{8}$$
$$2^{-4} = \frac{1}{16}$$
$$2^{-5} = \frac{1}{32}$$
$$2^{-6} = \frac{1}{64}$$
$$2^{-7} = \frac{1}{128}$$

___	___	___	___	___	___	___	___	.	___	___	___	___	___	___	___
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}

Example

0.5
0.25
0.125
0.0625
0.03125

$(0.0001)_2 = (0.0625)_{10}$

$(0.10101)_2 = 0.5 + 0.125 + 0.03125$

$= (0.65625)_{10}$

$$\begin{array}{r} 50000 \\ 12500 \\ 03125 \\ \hline 65625 \end{array}$$

$[0,1) * 2 \longrightarrow [0, 2)$

Converting Decimal Fractions to Binary

Powers of 2:

$2^{-1} = \frac{1}{2}$

$2^{-2} = \frac{1}{4}$

$2^{-3} = \frac{1}{8}$

$2^{-4} = \frac{1}{16}$

$2^{-5} = \frac{1}{32}$

$2^{-6} = \frac{1}{64}$

$2^{-7} = \frac{1}{128}$

1. Let m be a number less than 1
2. Move left to right (from radix point)
3. Multiply m by 2. Set bit to the value in the ones place.
4. Reset m to the stuff after the decimal place.
5. Continue until m=0

This can only be 0 or 1

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}

Converting Decimal Fractions to Binary

Converting 0.84375 to Binary

$0.84375 * 2 = \underline{1}.6875$

$0.6875 * 2 = \underline{1}.375$

$0.375 * 2 = 0.75$

$0.75 * 2 = 1.5$

$0.5 * 2 = 1.0$

0

1. Let m be a number less than 1
2. Move left to right (from radix point)
3. Multiply m by 2. Set bit to the value in the ones place.
4. Reset m to the stuff after the decimal place.
5. Continue until m=0

leading zeros

0	0	0						.	1	1	0	1	1	0	0	0	0	0	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}				

trailing zeros

Check:

$$\begin{array}{r} 0.5 \\ 0.25 \\ 0.0625 \\ + \; 0.03125 \\ \hline 0.84375 \end{array}$$

Converting Decimal Fractions to Binary

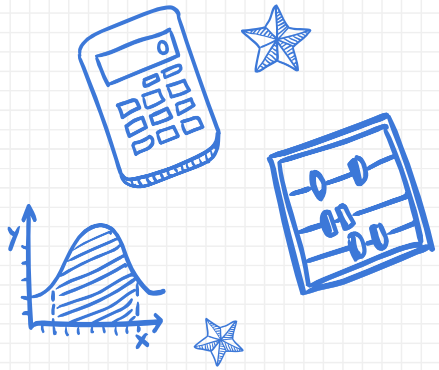
Converting 161.84375 to Binary

Convert the integer part and the fractional part separately, then stick them together with a radix point.

1	0	1	0	0	0	0	1	.	1	1	0	1	1		
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}

$1 + 32 + 128 = 161$

1. Let m be a number less than 1
2. Move left to right (from radix point)
3. Multiply m by 2. Set bit to the value in the ones place.
4. Reset m to the stuff after the decimal place.
5. Continue until m=0



Binary Arithmetic

Example

What is 30 in binary? What is 2 in binary? What is 32 in binary?

$(30)_{10} \longrightarrow (11110)_2$

check: $2+4+8+16 = 30$

$(2)_{10} \longrightarrow (10)_2$

$$
\begin{array}{r}
1\,1\,1\,1 \\
0\,0\,0\,0\,0\,1\,1\,1\,1\,0 \\
+\qquad\quad 1\,0 \\
\hline
\end{array}
$$

$(100000)_2 \longrightarrow (32)_{10}$

Adding two numbers in binary: Proceed right to left

In **decimal**: If the column exceeds **10**, we carry a 1 to the left

In **binary**: If our column exceeds **2**, we carry a 1 to the left

Example

Add $(11110)_2 + (10)_2$

Adding two numbers in binary: Proceed right to left

In **decimal**: If the column exceeds **10**, we carry a 1 to the left

In **binary**: If our column exceeds **2**, we carry a 1 to the left

Example

Subtract $(11101)_2 - (110)_2$

$$\begin{array}{r} 1\,\cancel{1}\,\cancel{1}\,0\,1 \; - \\ 1\,1\,0 \\ \hline 1\,0\,1\,1\,1 \end{array} \qquad \begin{array}{r} \cancel{1}0\cancel{7} \; - \\ 34 \\ \hline 3 \end{array}$$

Subtracting two numbers in binary: Proceed right to left

In **decimal**: We can take 1 from the column to the left, and bring 10 to the right.

In **binary**: We can take 1 from the column to the left, and bring 2 to the right.

$$(10)_2 \rightarrow (2)_{10}$$

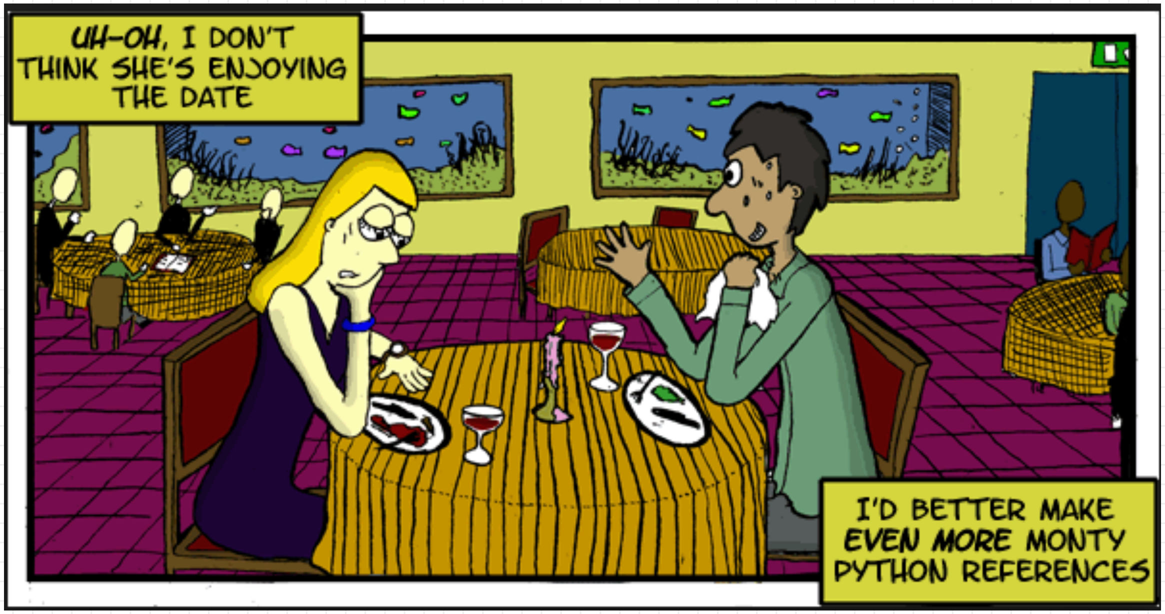
$$(1)_2 \rightarrow (1)_{10}$$

$$\begin{array}{r} (10)_2 \; - \\ (1)_2 \\ \hline (1)_2 \end{array} \iff \begin{array}{r} 2 \; - \\ 1 \\ \hline 1_{10} \end{array}$$

Intro to Python (Python 3)

Easiest way to get Python: https://www.anaconda.com/download

Good Practice: https://www.hackerrank.com/domains/python



More Examples IF WE HAVE TIME

"This calendar has 11 days a week, 9 weeks a month, and 17 months a year. I bought it at the Time Management Seminar!"

Example

Convert 0.1 from decimal to binary

Powers of 2:

$$2^{-1} = \frac{1}{2}$$
$$2^{-2} = \frac{1}{4}$$
$$2^{-3} = \frac{1}{8}$$
$$2^{-4} = \frac{1}{16}$$
$$2^{-5} = \frac{1}{32}$$
$$2^{-6} = \frac{1}{64}$$
$$2^{-7} = \frac{1}{128}$$

2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0 . 2^{-1} 2^{-2} 2^{-3} 2^{-4} 2^{-5} 2^{-6} 2^{-7}

Example

Powers of 2:

$$2^{-1} = \frac{1}{2}$$
$$2^{-2} = \frac{1}{4}$$
$$2^{-3} = \frac{1}{8}$$
$$2^{-4} = \frac{1}{16}$$
$$2^{-5} = \frac{1}{32}$$
$$2^{-6} = \frac{1}{64}$$
$$2^{-7} = \frac{1}{128}$$

Convert 0.1 from decimal to binary

Example: Convert 0.1 from decimal to binary.

0.1*2=0.2, so first bit is a 0

0.2*2=0.4, so next bit is a 0

0.4*2=0.8, so next bit is a 0

0.8*2=1.6, so next bit is a 1

0.6*2=1.2, so next bit is a 1

0.2*...NOWHOLDONASECOND!

- Restarts the pattern from here.

$0.1_{10} = 0.00011001100110011..._2$

___	___	___	___	___	___	___	___	•	___	___	___	___	___	___	___
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}

Example

$0.1_{10} = 0.00011001100110011..._2$

Powers of 2:

$$2^{-1} = \frac{1}{2}$$
$$2^{-2} = \frac{1}{4}$$
$$2^{-3} = \frac{1}{8}$$
$$2^{-4} = \frac{1}{16}$$
$$2^{-5} = \frac{1}{32}$$
$$2^{-6} = \frac{1}{64}$$
$$2^{-7} = \frac{1}{128}$$

- So computers would need to store an infinite number of bits in order to store 0.1_{10} exactly.
- ... obviously, computers can't do that.
- They truncate at a certain point, leading to some error.
- **How bad can this error be?**

2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0 . 2^{-1} 2^{-2} 2^{-3} 2^{-4} 2^{-5} 2^{-6} 2^{-7}

Example: Truncation error.

$$f(a,b) = 333.75b^6 + a^2(11a^2b^2 - b^6 - 121b^4 - 2) + 5.5b^8 + \frac{a}{2b}$$

where $a = 77617, b = 33096$

$z = 333.75b^6 + a^2 (11a^2 b^2 - b^6 - 121b^4 - 2)$

$x = 5.5b^8$

$y = z + x + a/(2b)$

$z = -7917111340668961361101134701524942850$

$x = \quad 7917111340668961361101134701524942848$

$\quad z + x = -2 \Longrightarrow y = -2 + a/(2b) = -0.827396 \ldots$

But, if precision $p \leq 35$, then

$\quad z + x \cong 0 \quad \Longrightarrow y \cong \qquad (a/2b) = 1.1726 \ldots$

Not even the correct sign!