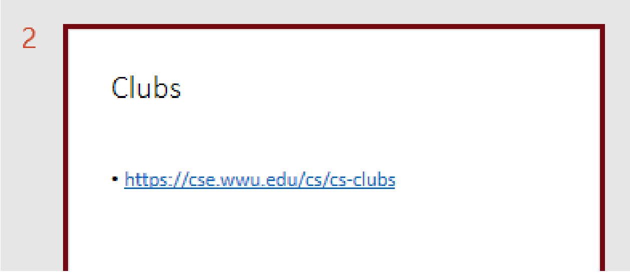
3: data types, Binary

Sunday, January 9, 2022 2:02 PM

Announcements

CS Clubs page https://cs.wwu.edu/clubs-weekly-schedule

https://cs.wwu.edu/cs-mentoring-program



Clubs

- https://cse.wwu.edu/cs/cs-clubs

To do for future: cover modulus, string concat, etc for A1

Hardware and storage

Thursday, January 13, 2022 10:09 PM

RAM (Random access memory)

In other words it is a white board where you grad a random spot to work

Close Thonny, open thonny

Notice our variables are empty again. only in RAM (but we saved our code to the HDD)

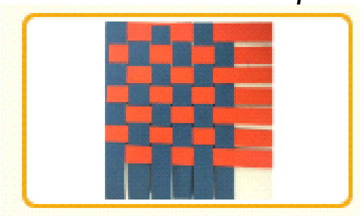
RAM Saves with electricity
fast, Power off = wiped

Hard drive (vs floppy (which was literally floppy))

Saves using magnetic states + −
or electron traps electron
No electron

Only 2 States!
.... How get Numbers like 53
or letters... (soon!)

History



Weaving is taking threads and going over, under (two states)

7

But if you don't just go 'over, under' but do things like,
'over over over, under, over over over' you can get interesting patterns
….and even super intricate art!

8

On old looms this was a huge hassle

55

Punchcard Looms

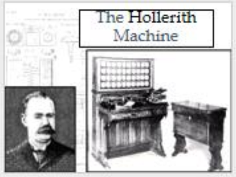
Then in 1745, Basile Bouchon invented a way to control the 'over, under' with paper tape, like one of those old player pianos. In 1805 Jacquard took Bouchon's fussy process and improved it, using cardboard punchcards, which were both more durable and more customizable

56

Jacquard
1804

"This portrait of Jacquard was woven in silk on a Jacquard loom and required 24,000 punched cards to create (1839). It was only produced to order. Charles Babbage owned one of these portraits ; it inspired him in using perforated cards in his analytical engine" Wikipedia, Sept 2011

57



The Hollerith Machine

Dr. Herman Hollerith built the first electrically-driven processing machine in order to accommodate the 1890 census. This machine used punched cards, not unlike those used until the late 70's. Hollerith's company eventually changed its name to International Business Machines…

58

First Electronic Digital Computer: 1939

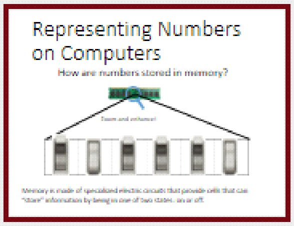
mechanical
electromechanical
electro digital

Binary

Thursday, January 13, 2022 10:19 PM

let's talk about Numbers

59

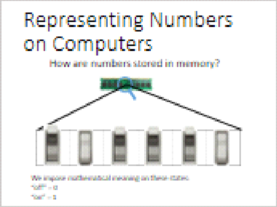


Representing Numbers on Computers

How are numbers stored in memory?

Memory is made of specialized electric circuits that provide cells that can "store" information by being in one of two states, on or off.

60

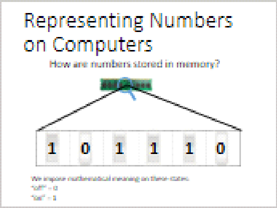


Representing Numbers on Computers

How are numbers stored in memory?

We impose mathematical meaning on these states.
"off" = 0
"on" = 1

61

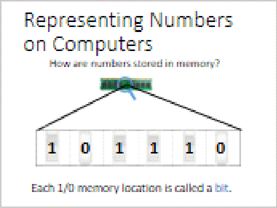


Representing Numbers on Computers

How are numbers stored in memory?

We impose mathematical meaning on these states.
"off" = 0
"on" = 1

62

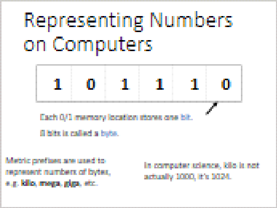


Representing Numbers on Computers

How are numbers stored in memory?

Each 1/0 memory location is called a bit.

63



Representing Numbers on Computers

1	0	1	1	1	0

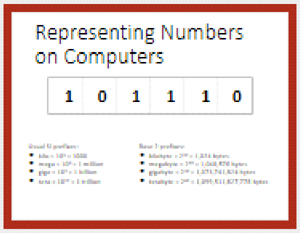
Each 0/1 memory location stores one bit.

8 bits is called a byte.

Metric prefixes are used to represent numbers of bytes, e.g. kilo, mega, giga, etc.

In computer science, kilo is not actually 1000, it's 1024.

64



Representing Numbers on Computers

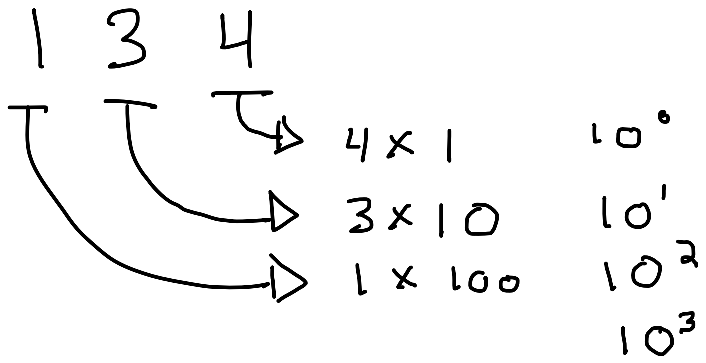
1	0	1	1	1	0

Base 10 prefixes:
- kilo = 10^3 = 1000
- mega = 10^6 = 1 million
- giga = 10^9 = 1 billion
- tera = 10^{12} = 1 trillion

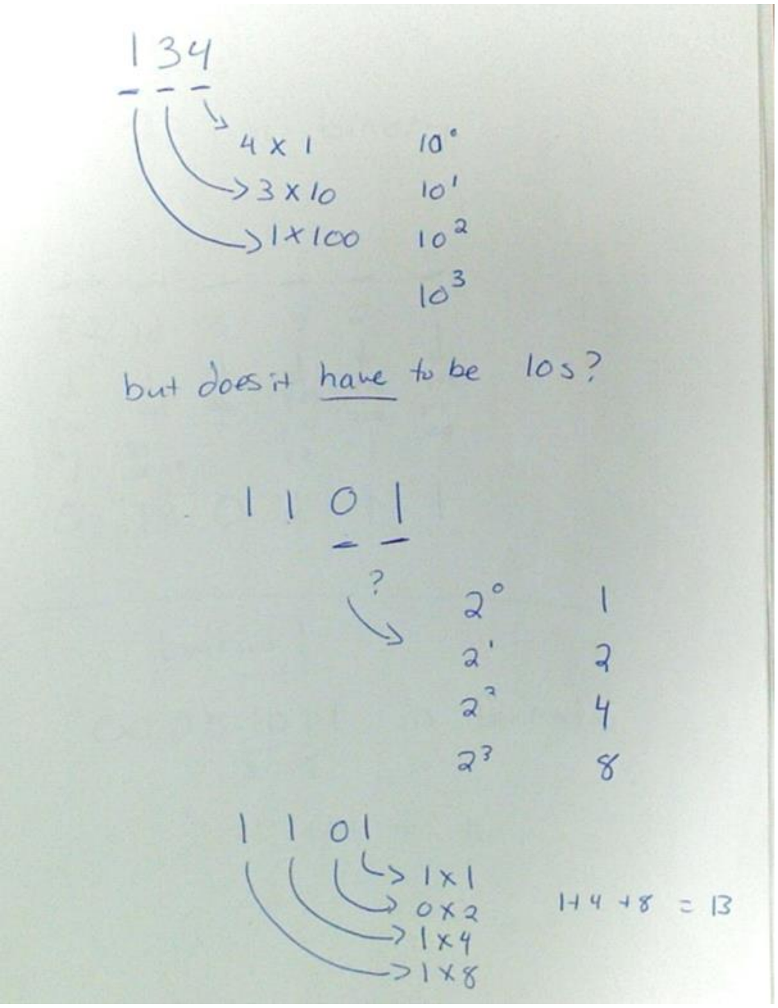
Base 2 prefixes:
- kilobyte = 2^{10} = 1,024 bytes
- megabyte = 2^{20} = 1,048,576 bytes
- gigabyte = 2^{30} = 1,073,741,824 bytes
- terabyte = 2^{40} = 1,099,511,627,776 bytes

Ok but how store
134 if you only
have 1 and Ø?

Breakout room:
w/o looking it up
brainstorm ways



Does it have to be 10s?



134

4 x 1	10^0
3 x 10	10^1
1 x 100	10^2
	10^3

but does it have to be 10s?

1 1 0 1

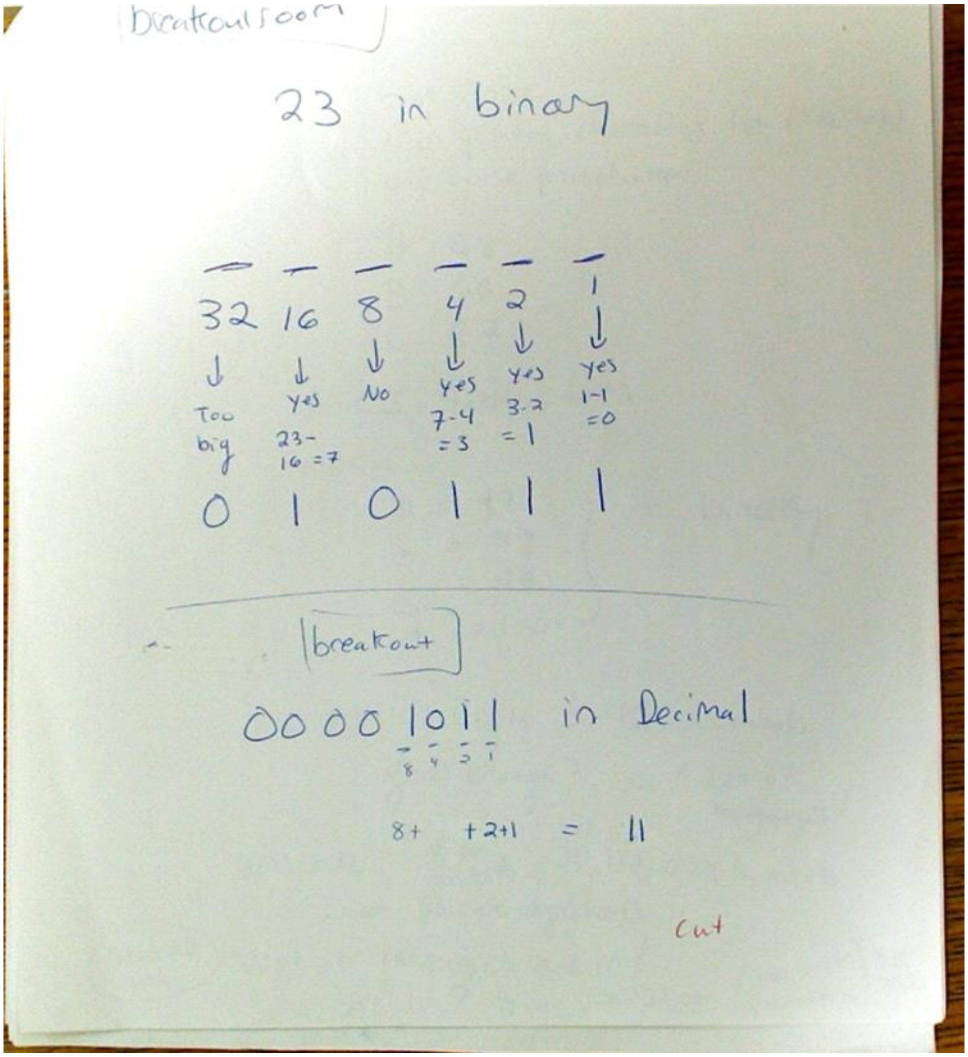
?

2^0	1
2^1	2
2^2	4
2^3	8

1 1 0 1

1 x 1
0 x 2
1 x 4
1 x 8

1+4+8 = 13



breakoutroom

23 in binary

32	16	8	4	2	1
↓	↓	↓	↓	↓	↓
Too big	Yes 23-16 =7	No	Yes 7-4 =3	Yes 3-2 =1	Yes 1-1 =0
0	1	0	1	1	1

breakout

0000 1011 in Decimal

8 4 2 1

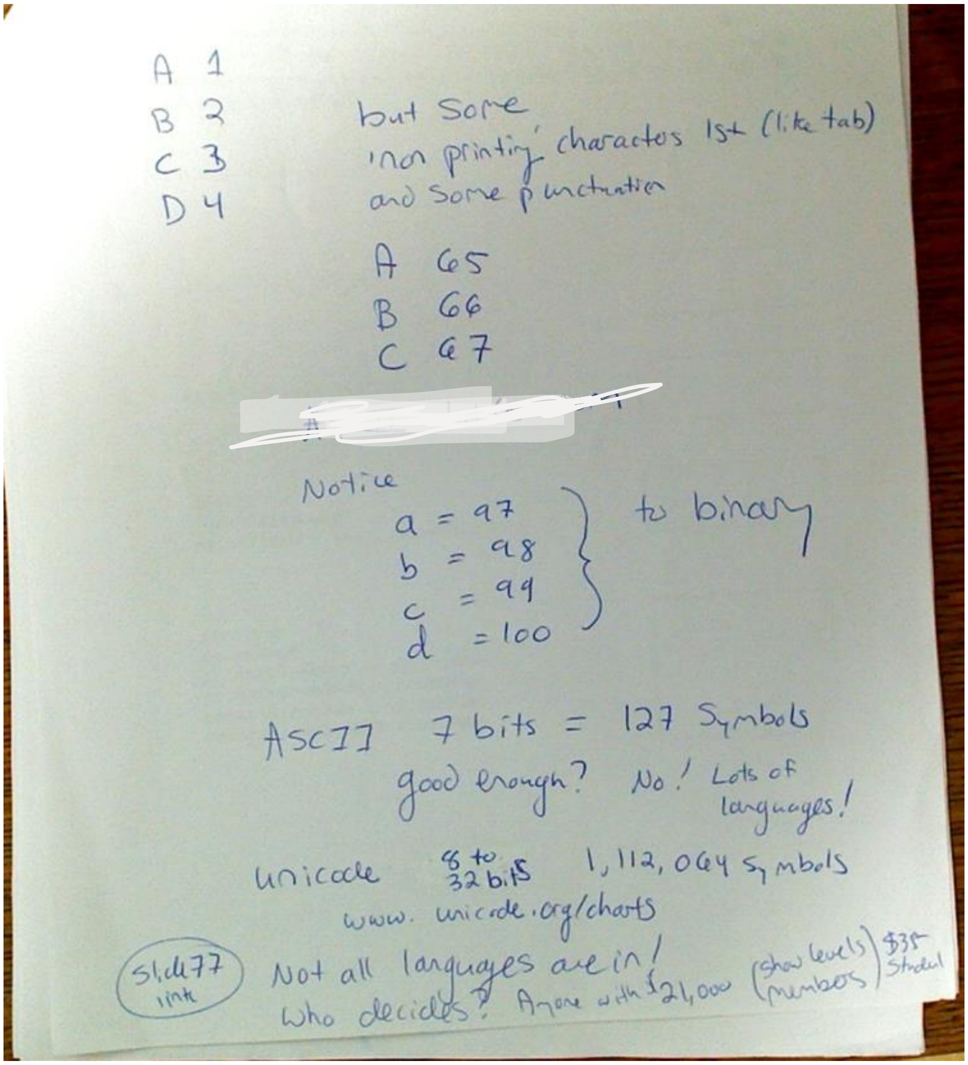
8+ +2+1 = 11

Cut

What about letters? Talk to the person next to you: how would you represent letters in binary?

It's all about agreeing that this pattern in binary represents something: a number, a letter, whatever!

https://bournetocode.com/projects/GCSE_Computing_Fundamentals/pages/img/ascii_table_lge.png



A	1
B	2
C	3
D	4

but some
'non printing' characters 1st (like tab)
and some punctuation

A	65
B	66
C	67

[redacted]

Notice

a = 97
b = 98
c = 99
d = 100

to binary

ASCII 7 bits = 127 Symbols

good enough? No! Lots of languages!

unicode 8 to 32 bits 1,112,064 symbols

www. unicode.org/charts

Slide77 link

Not all languages are in!
Who decides? Anyone with $21,000 (show levels members) $35 Student

Why isn't A 1? Well, when they invented ASCII people were still looking at raw binary,
And they wanted the control characters to really be easy to read

But notice A and a and 1 all end in 1 …so they tried
But letter 7 and number 7 are not the same!

https://home.unicode.org/membership/membership-levels/

Data Types and Variables

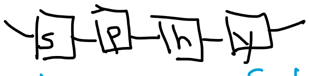
Sunday, January 9, 2022 2:23 PM

Integers 8 420 -1

Decimal 1.75 -0.45

AKA "float" ↑ Floating point

Strings "sphynx of black quartz, hear my vow" ↑ required



like a string of letter beads

Why? (more in CSCI 301)

"One" + 1 No can't do this

different types behave differently in Calculations and Storage

⇓

Print("hello world")

= input ("what do you wish")

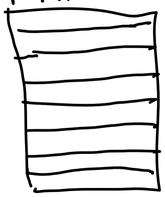
↑

Save to RAM

Pronounced 'puts'

(Save to file on HDD later in 141)

RAM



Space to Save many things

each spot gets an address

0x8048d28

↑ Not human friendly

Wish = input("name?")

Variable name (instead of memory address)

exceptions in future classes

uses_wish = input("wish:")

why called variable? Because contents can vary, or change. Lets say we decide the user should want a Pony

user_wish = "pony"

RAM

user_wish → 0x80dae57 → ~~"accordian"~~ "Pony"

```
user_wish = input("Your wish?")
user_wish = "pony"


user_wish = 1000000
```

Show variable

Show heap

show step by step debugger

Can we do this?

Yes → two types string & integer

Python is a dynamically typed language

Some programming languages are static type

Static typed means if you make a variable save a string, You can't later make it save a integer. why?

different data types actually need different amount of space to store in RAM & HDD (more next class!)

Think of Amazon packages:

using custom boxes wastes less but takes more time & attention

using 1 size box for everything is quick & easy but wastes space

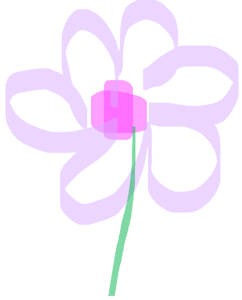
Write;

What type:

answer = 4

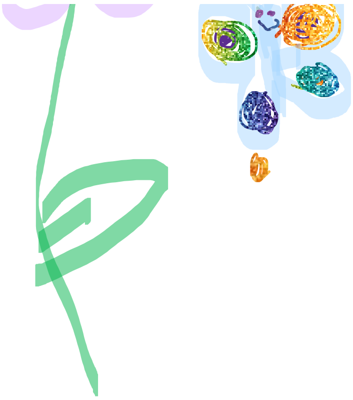
answer = "a pony! 5 ponies!"

answer = 99.9



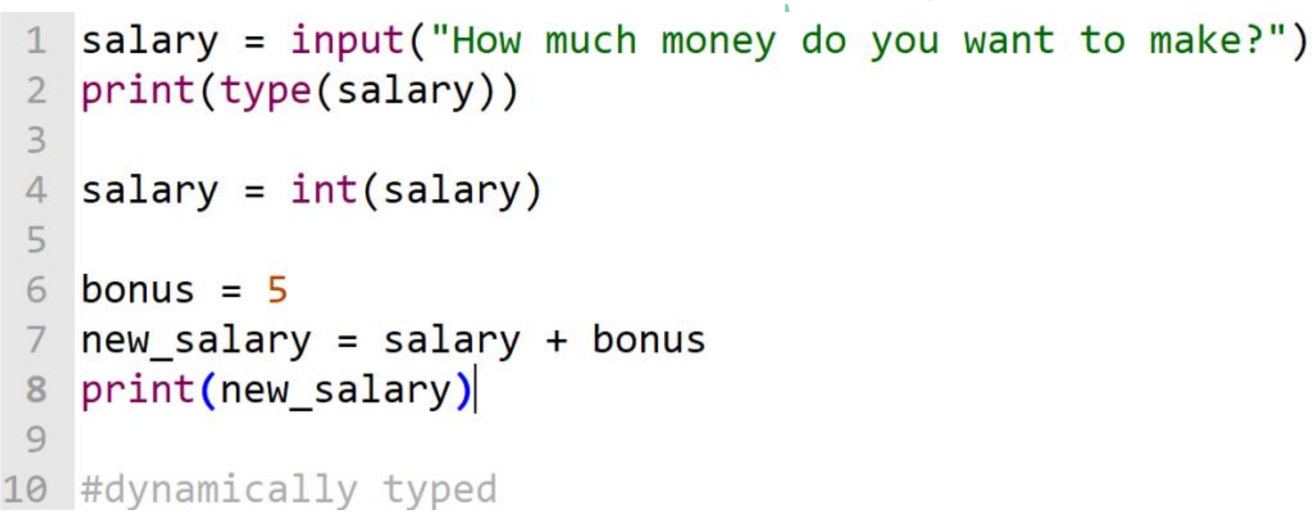
answer = 99.9

So whats in answer?



print(type(answer)) #demo this between each change

Answer = "4" #one, yes caps means different variable! Two, "4" vs 4



```
salary = input("How much money do you want to make?")
print(type(salary))

salary = int(salary)

bonus = 5
new_salary = salary + bonus
print(new_salary)

#dynamically typed
```