Sheaff

Nicholas LaJoie
ECE 331
2/7/17

* Use fgets(), not gets()
  (size arg)
  ⇒ getline works

* if ( ) {
  } else {   ← !!!

* cat xyz | ./re '\d+' ⇒ ./re < ~~valid~~ xyz

Validate lines similar to:

$AKSSD, 1944, 336, . . .

^\$AKSSD,(\d+,){2}(-?\d+,){2}(\d+,){3}(-?\d,){4}\d+,0,(-?\d+)$

in Perl:

```
#!/usr/bin/perl        , script (#!) followed by interpreter
while (<STDIN>) {
  next if (/ regexp /);      # / operater for regexp.
  print; //prints invalid    # if true, do next iteration
                             (can put that in front of if if only one statement)
```

"continue"

PERL: don't program like c!

Calculator →

```
#!/usr/bin/perl
print eval $_,"\n" while(<STDIN>);
```

⇓

default variable: $_

(STDIN assigned to $_)

* argv[0] = first arg passed!
* $0 is command ↳ $ARGV[0]

what type??

- Perl has 3 types:
  - Scalars → $x
  - Lists → @s
  - Hashes → %h

Perl tries to interpret your intent

"context"

* Perl is forgiving
  ⇒ use warnings for ease of mind

vi hex

sandbox

* lists are like arrays

```
@s = (x, y, z, 3, $_);
*indexing: $s[1] = y  //cause it's a scalar!
push @s, "SANTOS"; //put SANTOS @ end of list
```

man perldata

#Enviroment Variables are key-value pairs

(hashes)

* processing hex data

__attribute__((packed)); //at end of struct - prevents padding

* AVR accesses byte-by-byte

$ CAT XYZ | ./re '\d+'

XYZ

N:cur

→ 1

IT'S. The end of the world as we know it

→ XYZX9 3 abc

C STDIN? getline