Lecture 4

this language is now “turing” complete.

we can write any program now. but it would be really hard to do that.

because we don’t have:

1. decomposition-to break the codes into modules
2. abstraction-to suppress the details

one of the ways to do this is: FUNCTIONS

* by functions we can make new primitives
* it will take a input
* and give us an output as per whatever it does

How to define function?

def name(x) --this defines the formal parameters

return – keyword -- it basically says that when you get to this point and then return to the main control

None –special value and it has odd behavior:

* it’s a value
* that says that there is no actual value that has been returned

Invoke a function: by passing in values for the parameters: like name(t)

all variables declared in the functions are local and these don’t affect any global variables with the same name

in a function we can write a specification using 3 quotes. and while invoking it the specification will be there on the screen.

The farmyard problem:

he has bunch of pigs and bunch of chickens. he sees 20 heads and 56 legs.

numP+numC=20

4\*numP+2\*numC=56

Solving:8 pigs and 12 chickens

for computers we can start by checking each combination

0 and 20

1 and 19 and so on

we can return more than one value as a tuple

then while assigning too we must use a tuple

Recursion:

* the base case
* inductive step

isPalindrome: base case is size is 0 or 1