CS61a

1. Python Doesn’t print NONE in a Boolean
2. Python will print whatever value it reached before it terminated for Booleans
3. Print(‘x’) prints just x
4. Print(func) prints the functions function
5. Return(‘x’) will print ‘x’
6. Note: write if x>3: return T else: return F as ::: return x>3
7. A while loop never without a return statement wont go to any other code, and will return None once it terminates.
8. A while loop doesn’t repeat the whole code just itself
9. ‘x’ prints ‘x’
10. “x” prints ‘x’
11. “’x’” prints “’x’”
12. print(‘x’) prints x (Also functions)
13. You can start code with while and for
14. Str(x) return x as a string
15. eval(“x”) return what x is // x must be a string
16. While x: and while x>0: are equal // if x is decreasing
17. Int(x) changes from float to int rounding down
18. range will take values up to but not including your endpoint
19. Range is a sequence but it is NOT a list
20. Range(4) = range(0,4)
21. Global x :Define a value globally

NOTES on Print:

1. print(x,y) = x (space) y // any x and y
2. print(x+y) = x (NO space) y // x and y both same type
3. You can print a sequence
4. In a while loop, just use ONE print statement (e.g. hailstone)
5. Print(x) where x returns nothing literally prints None

Lists:

1. negative indicies counts backwards
2. indexed slicing doesn’t evaluate the end result
3. syntax LIST[start:stop]
4. index starts at 0
5. **len(x):** Return the length (the number of items) of an object.
6. X in y returns a Boolean// for int x and list y
7. In executing a line of code as in Disc 1 ; Q2.3 : 2 , the list will be bound to the last value it occupies.
8. You can return a list ; it will print itself
9. You can add lists together.
10. and multiply them with digits to increase their size
11. You can write this to select the ith element of a list (sequence) x = list\_name[index] // index is your input variable f(index) or your increment
12. A for statement does NOT introduce new frames
13. Sequence unpacking works only for fixed –length sequences; e.g. for x, y in list:
14. List(sequence) is a list constructor that takes in a sequence and spits out a list of the elements of the seq.
15. –ve index = len(list)-index
16. Check if an item exists in a list with **in**
17. Lst.remove(x) to remove x in lst from lst
18. Lst.pop /lst.delete(n) to delete indexed n x in lst
19. List.insert(index, x) add x in position index
20. [a[i] for i in (1,2,5)] to extract elements 1,2,5 (indexed)
21. l = [[1,2,3],[4,5,6]] to flatten a list of sublists
22. flattened\_l = [item for sublist in l for item in sublist]
23. s = reduce(lambda x, y: x + y, l) (from functools)
24. to sum a list: sum(l) l=list

**List Comprehension:**

1. Use any argument for comprehension except for the exact list. E.g. (bad example) [l for l in l if l in l]

**List slicing:**

TIPS:

1. python3 –i file.py to execute code
2. command K to clear
3. ctrl C to stop looping

What is an object?

* Rules for evaluating primitive expressions?

What is an iterable thing?

What is repr?

What is 2,3 in a mean? // Disc. 1 Q2.3

How to execute code in terminal?

Types of range?