Time

import time

def c\_to\_f(c):

return c\*9/5 + 32

t0 = time.clock()

c\_to\_f(100000)

t1 = time.clock() –t0

print("t = ", t1, "s, ")

Operations

def c\_to\_f(c):

return c\*9/5 + 32

def mysum(x):

total = 0

for I in range(x+1):

total += 1

return total

Inputs

def search\_for\_elmt(L, e):

for i in L:

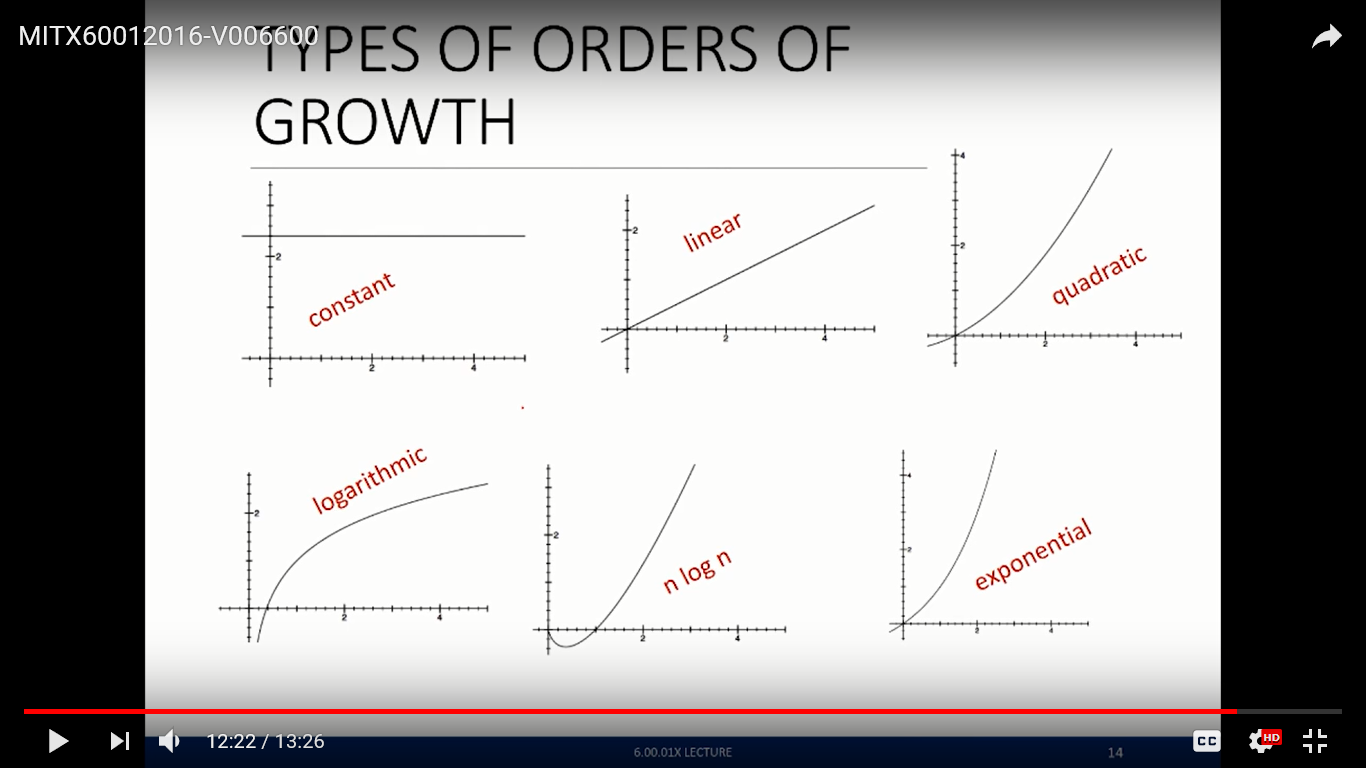
if i == e:

return True

return False

[https://www.youtube.com/watch?v=RJhZ5SD1AvY](https://www.youtube.com/watch?v=RJhZ5SD1AvY" \t "_blank)

Types of Growth



Big O Notation

Measures an upper bound on the asymptotic growth, also called order of growth

def fact\_iter(n):

"""assumes n is an int >= 0"""

answer = 1

while n>1:

answer \*= n

n -= 1

return answer

: n2 + 2n + 2

: n2 + 100000n + 31000

: log(n) + n + 4

: 0.0001\*n\*log(n) + 300n

: 2n30 + 3n

Exercises

def linearSearch(L, x):

for e in L:

if e == x:

return True

return False

Choose which of the following inputs to linearSearch would give the best case, average case, or worst case run time.

1. Best Case Run Time

linearSearch([14, 15, 6, 27, 13, 16, 25, 11, 7], 15)

linearSearch([21, 1, 25, 22, 30, 13, 7, 24, 12], 24)

linearSearch([20, 10, 1, 7, 4, 22, 25, 12, 31], 20)

linearSearch([9, 3, 12, 24, 7, 8, 23, 11, 19], 8)

linearSearch([4, 12, 20, 17, 9, 14, 7, 24, 6], 7)

linearSearch([13, 9, 22, 3, 10, 17, 11, 2, 12], 26)

1. Worst Case Run Time

linearSearch([14, 15, 6, 27, 13, 16, 25, 11, 7], 15)

linearSearch([21, 1, 25, 22, 30, 13, 7, 24, 12], 24)

linearSearch([20, 10, 1, 7, 4, 22, 25, 12, 31], 20)

linearSearch([9, 3, 12, 24, 7, 8, 23, 11, 19], 8)

linearSearch([4, 12, 20, 17, 9, 14, 7, 24, 6], 7)

linearSearch([13, 9, 22, 3, 10, 17, 11, 2, 12], 26)

1. Average Case Run Time

linearSearch([14, 15, 6, 27, 13, 16, 25, 11, 7], 15)

linearSearch([21, 1, 25, 22, 30, 13, 7, 24, 12], 24)

linearSearch([20, 10, 1, 7, 4, 22, 25, 12, 31], 20)

linearSearch([9, 3, 12, 24, 7, 8, 23, 11, 19], 8)

linearSearch([4, 12, 20, 17, 9, 14, 7, 24, 6], 7)

linearSearch([13, 9, 22, 3, 10, 17, 11, 2, 12], 26)

1. What is the number of steps it will take to run linearSearch in the best case? Express your answer in terms of *n*, the number of elements in the list L.

Indicate addition and multiplication explicitly, with + and \* symbols. Indicate exponentiation with the caret (^) symbol.

1. What is the number of steps it will take to run linearSearch in the worst case? Express your answer in terms of *n*, the number of elements in the list L.

Indicate addition and multiplication explicitly, with + and \* symbols. Indicate exponentiation with the caret (^) symbol.