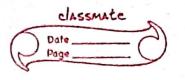
	CLA	ssmate.	Of the last
5	Date_	The state of the s	# CONTRACTOR
C	Page_		

SORTING RECURSIVELY

```
CODE :
     4 Moto: Remarker notes (detailed) in treks
def mini (L):
   # finds the minimum element in the list
    for a in L:
      if (24 mini):
     Pergraminica tel a ni O boit : margarel
    return mini
def Sort (L): (0 = (stant) to

# recursively sorts the list and much
    if (L==[]) or (len(L)==1) 0 = 10] 1 1
    # if the list is empty there is nothing to sost
    m = mini (L)
   #m now contains the minimum most element in L
   L. remove (m)
   # we semone that element from L
   seturn [m] + Sort (L)
   # we recursively sort the smaller list
L = [5, 6, 59, 19, 2, 7]
print (Sort (L))
OUTPUT: [2,5,6,7,19,59]
```



Binasy-Seasch

#	What is Binary Search?
-	Binary search is a searching algorithm for finding
	an element's position in a corted array.
-	In this approach, the element is always searched in
	the middle of a poetion of an array
-	Binary search can be implemented Jonly on a costed
	list of items Al H. Rolling to all soft a cost of its
	we need to work them first!
	mid of the second
111	· T. of x = rold. then seturn roll. Else unpose
#	BINARY SEARCH WORKING
2 -	Binary search algorithm can be implemented in two
164)	ways which are . which it is it is
	I have by entiry low to have - while!
	Iterative Method
(2)	Recursive in the the state of the surgence and all . I
ن	(The recurire method follows the directed conquer approach
-	en selling high to high mid-
	The general steps for both the methods are:
	[3, 4, 5, [4], 1, 1, 1]
1.	and the property of the performance
	[3,4,5,6,7,(8,9)]
	Let c=4 be the element to be searched!

Just.

2=4 is found. 3, 4, 5, 6, 7, 8,9] ITERATIVE METHOD CODE: def binary search (L, k): # we want to shrink our list # we will do that by using while loop begin = 0 # first element in L, L[0] end = len (L)-1 # last element in L, [len(L)-1] # using a while loop to look at the list and keep halving it While (end-begin > 1) # we will handle the case when the no. of elements # less than or equal to 1. mid = (begin + end) //2 if (L [mid] = = K): # if mid is indeed to, then we H ketum True & stop the code. return True if (L[mid] > k): # if mid element is greater than k, H we will check on the left side end - mid -1 if (L[mid](k): begin = mid+1 # if mid element is less thank, # we will check on the right eide

	classmate
Q	Dafe

This is exclide while loop. If we are here, it means # that we haven't found the element. Also, if we are # here, it means that the while condition is violated. # Which means end-begin is less than or eghal to

if it is equal to 1, then there are exactly a climental

if (L[begin]==k) or (L[end]==k): H(0) else: (d. 21 draw prairied) de

don't like prive ut to the they wast

def shinary search (L, k, begin, end) : 1 "This will recurringly compute binary search # if begin and end are same, then we need to

just Check L[begin] I d. Soud - box balletes

if (begin == end): if ([lugin] == k): return True

if begin and end are consecutive, then check # them individually.

if (end-legin == 1():

4 ([[begin] == k) or ([[end] == k): Return True

	classmate
	Date
che:	
etum Fal	
Lelium Fal	le
if (end-begin 71):	
A compute the midd	le element
mid = (begin - end	
if (L[mid]>k):	
	ght betain the left
end = mid-	Jac - Lange San Cape
if (L[mid] (k):	
the discoult the contract of t	14 0 412-41 33 14
# discard the C	if & retain the light
begin = mid + 1	data ili a non-ileo di
if (L(mid) = = k)	
Leturn True	/ A 1
et net leger white for the line	1 11
if (end-begin <0)	is it per a take att 100
return False	reducire
some or exist. So a file se	lunger at transacration with
return rbinary seard	r (1, te, begin, end)
	as the als of the short
· los zut	are the it the british
A STATE OF THE STA	V. V.
the objection the war in the	Horse in Pulling is I
	I has sain all all
	1. Chron in Lile
m oresalies)	
1 201135 2316 31	The state of the s
	Light states - E
	Water a state of the state of t
ALTO SER	and the manufacture of the same of