Assignment os Sorting Algorithms - Insertion Sort, Shell Sort (55-151) # Problem Statement write a python program to store second year percentages of students in an away write function for sorting array of floating point numbers in ascending usder by shell sort and display top five scores. # Objectives: isto understand working of insertion sort & shell sort algorithm. 2) To able to use let in python to implement this algorithmy. # Soffware requirement: Operating System: windows 10 house, single language Python version: 3.8.5 us code (bext editor). Sept 2020 Version. # Hardware requirement; in book sold shall Manufactures: Acer PROCESSOR: Intel (P) Core 15-8265 U CPU @ 1.66H2. 1.86H2. System type: 64-bit operating system, 2-64 based processos # Theory: 20tha described that Employers the Base Want a) Insection Set: In this, array is sorted by insecting ask element from unsorted part to saled part one at a time. Entrally we have all elements in unsorted part. The algorithm works by inserting element in sorted

Date Page

part, have it is called inserting sort algorithm.

by shell feet:

y It is a modification of insertion sort algorithm.

in The insertion sort algorithm is improved to work

botter in some cases, of this is called shell-sort.

In The sholl sort works by breaking the original list

into number of smaller subjists after which each

ene is sorted using insertion sort.

Instead of breaking list into subjist of contiguous

item, it was an increment which is called gap

to create subjist by choosing all items that due i

apart.

Pseudo lode:

in Insection Sort:

Atgorithm Insection (List, 4):

for it to in:

val + list cm

holexi

while [hole>1 and Irsb [hole+] > val)

luf [hole] = | 136 Chole+]

hole + hole -1

but [hole] = val.

27 shell sort (waring gap will be halved refer each pass)

Algorithm Shell Sort (list, n)

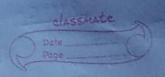
gap < 1/2

while (gap>0):

j=gap

while (j<n):

1= j- gap



if (lust [itgap) < lust [i]): Swap (lust [itgap], lust [i]) ADT for classis class Scoze 8heef: def constructor-functiones: Minitralize empty list lush = [], u=0 000 def invertion Sorb (): - 11 lugic of insertion sort def shell soit (): 11 logic of shell soit to love 15 4 90.22,85-23, 18 11 paints top 5 stidents marks. Teme & space complexity Analysis. Insection Estate 1 10.00 postporase Space complexity: The algorithm doesn't require

gry additional space. Hence it's a constant

space algorithm. Space complexity: O(N).

Time complexity: Best case: In case of sorted array; O(n) worst case in case of severse sorted array; O(4°) time complexity of insertion sort is o(4°)

& Shell sort

- space complexity: The algorithm doesn't orquore addition space of our space is orquored to store the assay.

Time complexity: The time complexity of algorithm to bold but the algorithm will have less number of comparison compared to insertion sort it is a improvement over insertion sort.

Testicases

1	a supposed to treatment of the first of the					
	Testiage no	Testicase Percription	EXP	perted o/p	Actu	al opp.
	1.	Susertion sort (No	ank	marks	Roule	marks
		duplicatos)	1	-96.28	1	96.28
	No. of the least	lof= 790.22, 85.23, 96.28,	2	93.26	2	93.26
		53.55, 82.91, 93.26	3	90.22	3	90.22
			4	85.23	14	85.23
			2	82.98	1.5	82.91
	2.	Shell Soil (No duplicates)	Rouk	. marks	Euk	marteo
		lust= 190.22,85.23,	1	96.28	1	96.28
-		96.28,53.55,82-91,93.26	2	93.26	2	93.26
1		edron students ? a	3	30.22	3	90.22
-		REMARKS THE RES	4	85.23	4	85.23
1		index representation	5	82.91	5.00	82.9)
		Insection Sort (with dipli)	Rock	marks	Rank	marks
	,	luf: (90.21, 88.26, 90.21,	1	93.83	Salethie	93.83
6	wineach h	92.53, 88.26, 93.83}	2	92.53	2	92.53
	D STITE	with song longither	300	30.2)	3	90.21
	10 saylissign	of sure sure 10	9 -	88.26	4	88.26
	4.	Shell Soit (with duplicates)	Rank	marks .	Rank	marks
	Maril	uf = (90.21,88.26, 90.2)	10	93.83	129	93-83
	10 min 5	32.53, 88.26, 93.83	2	92.53	2	92.53
	N10 12 1	ans a martypian . An .	3	30.2	3	90.2
			9	88.26	9	88.26



Applications:

The injection soit & shell soit are one of the emplest sating algorithm.

They can be used to sort data in complicated scenarios where olulogu) sorting algorithms cannot be used.

Conclusion:

At the end of assignment I'm capable of implementing insertion soft shell set algorithm.

Also able to analyze space of time couplerity of

algorithms.