	Data Data	
	Assignment 5	
	Cher sequendally. I then awarded	
417-14	class SE IV DUDING SALE AL ROLL NO : 21430	
-	Batch = F4 10 portioni Dos: 22/03/2020	
100	Problem statement:	
	Write a python program to store second year	
	percentage of students in array white function	
_	percentage of students the control members	
	for sorting array of floating pointing members	
_	in ascending order using a) Insertion sort	
-	Shell sort ( ) Display top five scores!	
	insertion sort which works by break	
-12	Objective-invastal tri longio	
109	to leave and implement sorting algorithm of	
	shell sort and inscration sort.	Ī
		Ī
	Outcome: stai toil parasid is better	
130	Learnt to implement insertion sort and shell	
	sort to sort given floating numbers in ascending	J-
i	order using lists , loops and decision statement	
	Planents apart.	
	S/W requirement:	1
	Python IDE (pychairm) , Python bausa	_
		7
	H/W requirement: : the modern	_
	Laptop or PC	
	pserbion serte	1
	Theory:	-
	Insertion Soxt in The The	
	mention sort is sorting mechanism where	
	the sorted array is built having one item at	
7	a time . The elements are compared with each	

Pege No.  Date
Assignment 5
Other sequentially of then arranged simula
neously in some particular order. As it
works by inserting on element at a
particular position, it is known as insertion
Sort. : Jasastate asidora
White a pythes program to store second year
ii) Disertion sort in hole as application
in serting away of Floating printent members
- trii) shellesort to prize rabo pribasso ai
The shell sort is an improvement in
insertion sort which works by breaking
the original list into number of small
sublists after which each one is sorted
using on insertion sort on the lade
Insted of breaking list into sublist of
contiguous elements, it jusés en increment
i'i' which is called as gap to create a
sublist by choosing all items than are i
elements apart.
S/W requirement:
Psuedocode of Algorithm;
i. Insertion Sort: : Jasansviuper WIH
29 to got got
Insertion_sort()
for i=1 ton :- mont
Key = Self-percentage[i]
mention sort is sortified = prechanism where
to mel an privabile: jo= and ptil > key : 11
percentage [j+i] = percentage [j]

	Paga Mo.			
	Date			
_	END of while			
_				
	END OF Formi posses			
	ession ails	11 15		
	Sort data using snell sort			
11)	> psuedocode for shell sort			
_	END opine loop			
	shell-sort()			
	n = len (self, percentages)			
	gap = P/12	14	-	
	for i in range (gap, o): Wirely	0	-	(
	for i in range (gap,n):			
	for k in range (1-gap, -1):	at.	(1	
	+ case complexity = con	2.90		
	(mo visif (p[ktgap] > p[k]);	طم		
	(break; ixalgma)		2	
	clse		7	and the same
	Swap ( P[ K+gap], P[ K	1)	en	7-1-1
	(ni Fingapiques asso t	298		Marine L
	ist case complexity = of tilesof	اماد		
	Tabula divelamo	ACIO	3.	
	gap = gap(12(1)) : pixalques ?	1		
	END			
	6			
	Algorithm for main function:			
	Start			
2	create object of class student			
3.	. Display Menu to users			
4	I Input choice from user			
S	Tr choice==1:			
	Input data from user.			

-

	Page No.
	Date
	THE THE PARTY OF T
	elif choice == 2 insertion sort
	elif Choice == 2 sort data using insertion sort
	elif choic == 3
	elif choice== 4
_	elif choice
_	END while loop
4	else print ("privalid input")
1	print ("privation in
6.	END
	for i
_ *	Complexity: (0, 100) appropriate
	. (9,90)
is	prosting sort in sort in
	Best case complexity = 0 (n2) Worst/ average case complexity = 0 (n2)
	space complexity: O(1)
	space complement
-	We in a set of villa values
11)	shell sorte
-	Best case complexity = 0 (n)
_	Worst case complexity = 0 ( nlog n)
	space complexity: 0(1) 1988 = 918
	Algorithm for main funtion:
	Jyoth
	theorte object of class student
	ZOZU IL WASH VOLGER
	a toput donce from user
	i i sa such di
	topet data from user

*	Test cases			Page No.					
	Description	Input	Expected O/P	Actual 0/P	Result				
1	Enter Data	n = 3 [72.3,76.0, 90.4]							
2.	Disertion	n=3 [72.3,76.0, 90.4]	[723, 16.0 90.4]	[12.3,76.0	Pass				
3.	Shell		[12.3, 76.0, 90.4]	[72.3, 76.0 90.4]	Pass				
	Conclusion:  Students learnt the concept of insertion and shell sort and how to implement them.								

```
def insortt(p ):
                                                                                                                                                Analyzing...
           n=len(p)
           for i in range(1,n):
               temp=p[i]
               i = i - 1
               while j >= 0 and temp < p[j]:
                   p[j + 1] = p[j]
                   j -= 1
               p[j+1] = temp
           print("List is sorted by Insertion sort ",p)
       def shell_sort(p):
           n = len(p)
           gap = n // 2
           while gap > 0:
14
                for i in range(gap, n):
                    temp = p[i]
                    j = i
                   while j >= gap and arr[j - gap] > temp:
                        p[j] = p[j - gap]
                        j -= gap
                    p[j] = temp
               gap = gap / / 2
           print("List is sorted by Shell sort ", p)
        shell_sort() > while gap > 0
```

```
def top_five(p):
                                                                                                                                     A 12 ^ V
    i=-1
    print('\nTop five percentage : ')
    while(i>-6):
        print(p[i])
        i=i-1
arr =[]
n=int(input("How many elments do you want "))
for i in range (0,n):
    m=int(input("enter elemnt "))
    arr.append(m)
for i in range (0,n):
    print(arr[i]," ")
while(True):
    print("Enter Your Choice")
    print("1. Insertion Sort")
    print("2. Shell Sort")
    print("3.Top 5 Score")
    print("0. Exit")
    ch = int(input("Enter your choice : "))
    if(ch==1):
        insortt(arr)
    elif(ch==2):
```

```
m=int(input("enter elemnt "))
    arr.append(m)
for i in range_(0,n):
    print(arr[i]," ")
while(True):
    print("Enter Your Choice")
    print("1. Insertion Sort")
    print("2. Shell Sort")
    print("3.Top 5 Score")
    print("0. Exit")
    ch = int(input("Enter your choice : "))
    if(ch==1):
        insortt(arr)
    elif(ch==2):
        shell_sort(arr)
    elif(ch==3):
        top_five(arr)
    elif(ch==0):
        print('Thank You!')
        break
    else:
        print('Enter valid input')
```

```
d ⇒ How many elments do you want >> 5
   🛂 enter elemnt 🥕 14
     enter elemnt > 74
     enter elemnt > 65
     enter elemnt > 20
     enter elemnt > 99
     14
     74
     65
     20
     99
     Enter Your Choice

    Insertion Sort

     2. Shell Sort
     3.Top 5 Score
     0. Exit
     Enter your choice : > 1
     List is sorted by Insertion sort [14, 20, 65, 74, 99]
     Enter Your Choice
     1. Insertion Sort
     2. Shell Sort
```

```
Python Console × main × FDS ASSIGNMENT3 ×
                                      FDS ASSIGNMENT3 (1) ×
  ₹ 3.Top 5 Score
     0. Exit
     Enter your choice : >? 2
     List is sorted by Shell sort [14, 20, 65, 74, 99]
     Enter Your Choice
    1. Insertion Sort
     2. Shell Sort
     3.Top 5 Score
     0. Exit
     Enter your choice : >? 3
     Top five percentage :
     99
     74
     65
     20
     14
     Enter Your Choice
     1. Insertion Sort
     2. Shell Sort
     3.Top 5 Score
```

