Algorithm:

Algorithm for insertion:

1. Start
2. Repeat the following steps as many times as the length of the list L
   1. Set curr as element at position i in list L
   2. Set j as i-1
   3. Repeat the following steps while j is greater than 0 and the element at position j-1 in list L is greater than curr
      1. Set element at position j+1 in list L as the element present at position j in list L
      2. Decrement pos by 1
   4. Set the element at position j+1 as curr
   5. Print the list L
3. Return list L
4. Stop

Algorithm for shell:

1. Start
2. Set p as 0
3. Set gap as length of list L floor divided by 2
4. Repeat the following while gap is greater than 0
   1. Run a loop from i equal to gap to i equal to length of list L
      1. Run a loop from j equal to i-gap to j equal to -1 with decrement in j as gap
         1. If element at position j+gap in list L is greater than element at position j in list L, then
            1. Exit from loop
         2. Else
            1. Swap elements at positions j and j+gap in list L with each other
   2. Set gap equal to gap floor divided by 2
   3. Increment p by 1
   4. Print the list L
5. Return the list L
6. Stop

Algorithm for main program:

1. Start
2. Repeat the following
   1. Input from user the number of students in second year and store in a
   2. If a is greater than 5
      1. Exit from the loop
   3. Else
      1. Print(“Students should be greater than 5”)
3. Declare an empty list L and L1
4. Repeat the following steps for a times
   1. Repeat the following steps
   2. Try
      1. Enter the percentage of students and store in x
      2. If b greater than equal to 0 and less than equal to 100
         1. Append x in list L
         2. Append x in L1
         3. Exit from loop
      3. Else
         1. Print(“Not a valid percentage”)
   3. Except Value Error:
   4. Print(“Enter a number”)
5. Print the list L
6. Repeat the following steps
   1. Print(“Main Menu”)
   2. Print(“1.Insertion Sort”)
   3. Print(“2.Shell sort”)
   4. Print(“3.exit”)
   5. Take the input from the user and store in choice
   6. If choice is equal to 1
      1. Call intersection and store the list in op
      2. Reverse the list op and store in Li
      3. Print the list op to show scores in sorted order
      4. Repeat the following for 5 times
         1. Print the element at current position of list Li
   7. Else if choice is equal to 2
      1. Call shell and store the list in op
      2. Reverse the list op and store in Li
      3. Print the list op to show scores in sorted order
      4. Repeat the following for 5 times
         1. Print the element at current position of list Li
   8. Else if choice is equal to 3
      1. Print(“Thank you”)
      2. Exit from loop
   9. Else
      1. Print(“Please enter correct input”)

Pseudo Code:

Class sort

{

1. Insertion(L)
2. Sort(L)

}

Insertion(L)

{

1. For i=1 to i=len(L)
   1. Curr=L[i]
   2. j=i-1
   3. While(j>0 and L[j]>curr)
      1. L[j+1]=L[j]
      2. j=j-1
   4. L[j+1] = curr
   5. Print(“The list after iteration is”, L)
2. Return L

}

Shell(L)

{

1. P=0
2. Gap=len(L)//2
3. While(gap>0)
   1. For i=gap to i=len(L)
      1. For j=i-gap to j=-1 with decrement in j as gap
         1. If L[j+gap]>L[j]
            1. Break
         2. Else
            1. L[j+gap], L[j]=L[j], L[j+gap]
   2. Gap=gap//2
   3. P=p+1
   4. Print(“The list after pass is”, L)
4. Return L

}

Main Program()

{

1. obj=sort()
2. while(True)
   1. a=int(input(“Enter the number of students in second year:”))
   2. if a>5
      1. break
   3. else
      1. print(“There should at least be more than 5 students in second year”)
3. L=[]
4. L1=[]
5. For i=0 to i=a
6. try
   1. While(True)
      1. x=float(input(“Enter the percentage of student {i}:”))
      2. if(x>=0 and x<=100)
         1. L.append(x)
         2. L1.append(x)
         3. Break
      3. Else
         1. Print(“The percentage of a student must lie between 0 to 100”)
         2. Except Value Error:
         3. Print(“Enter a number”)
7. Print(“The entered percentages are”, L)
8. While(True)
   1. Print(“Main Menu”)
   2. Print(“1.Insertion sort”)
   3. Print(“2.Shell sort”)
   4. Print(“-1.Exit”)
   5. Choice=int(input(“Enter the sorting operation that you want to perform to find the top 5 scores”))
   6. If choice == 1
      1. a=obj.insertion(L)
      2. Li=a[::-1]
      3. Print(“The scores in sorted are:”, a)
      4. Print(“The top 5 scores are:”)
      5. For i=1 to i=5
         1. Print(“{i})”, Li[i])
   7. Else if choice == 2
      1. a=obj.shell(L1)
      2. li=a[::-1]
      3. Print(“The scores in sorted are:”, a)
      4. Print(“The top 5 scores are:”)
      5. For i=1 to i=5
         1. Print(“{i})”, Li[i])
   8. Else if choice == -1
      1. Print(“Thank You”)
      2. Break
   9. Else
      1. Print(“Please enter correct input.”)

}