Shell Sort

Sorting Data by Using Shell Sort

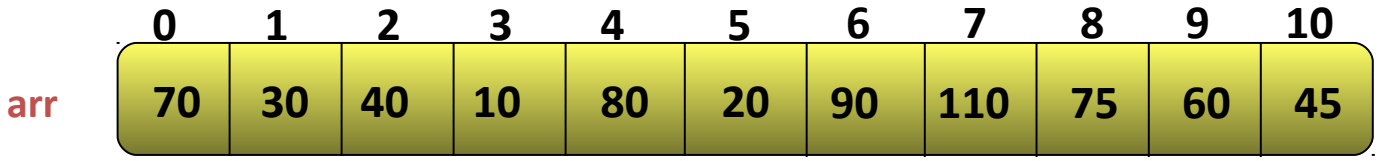
- Shell sort algorithm:
  - Insertion sort is an efficient algorithm only if the list is already partially sorted and results in an inefficient solution in an average case.
  - To overcome this limitation, a computer scientist, D.L. Shell proposed an improvement over the insertion sort algorithm.
  - The new algorithm was called shell sort after the name of its proposer.

Implementing Shell Sort Algorithm

- Shell sort algorithm:
  - Improves insertion sort by comparing the elements separated by a distance of several positions to form multiple sublists
  - Applies insertion sort on each sublist to move the elements towards their correct positions
  - Helps an element to take a bigger step towards its correct position, thereby reducing the number of comparisons

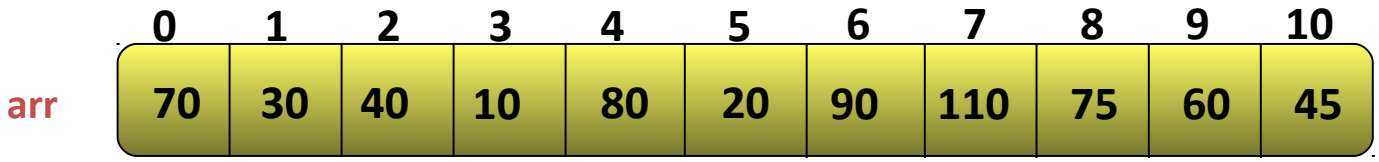
Implementing Shell Sort Algorithm (Contd.)

- To understand the implementation of shell sort algorithm, consider an unsorted list of numbers stored in an array.

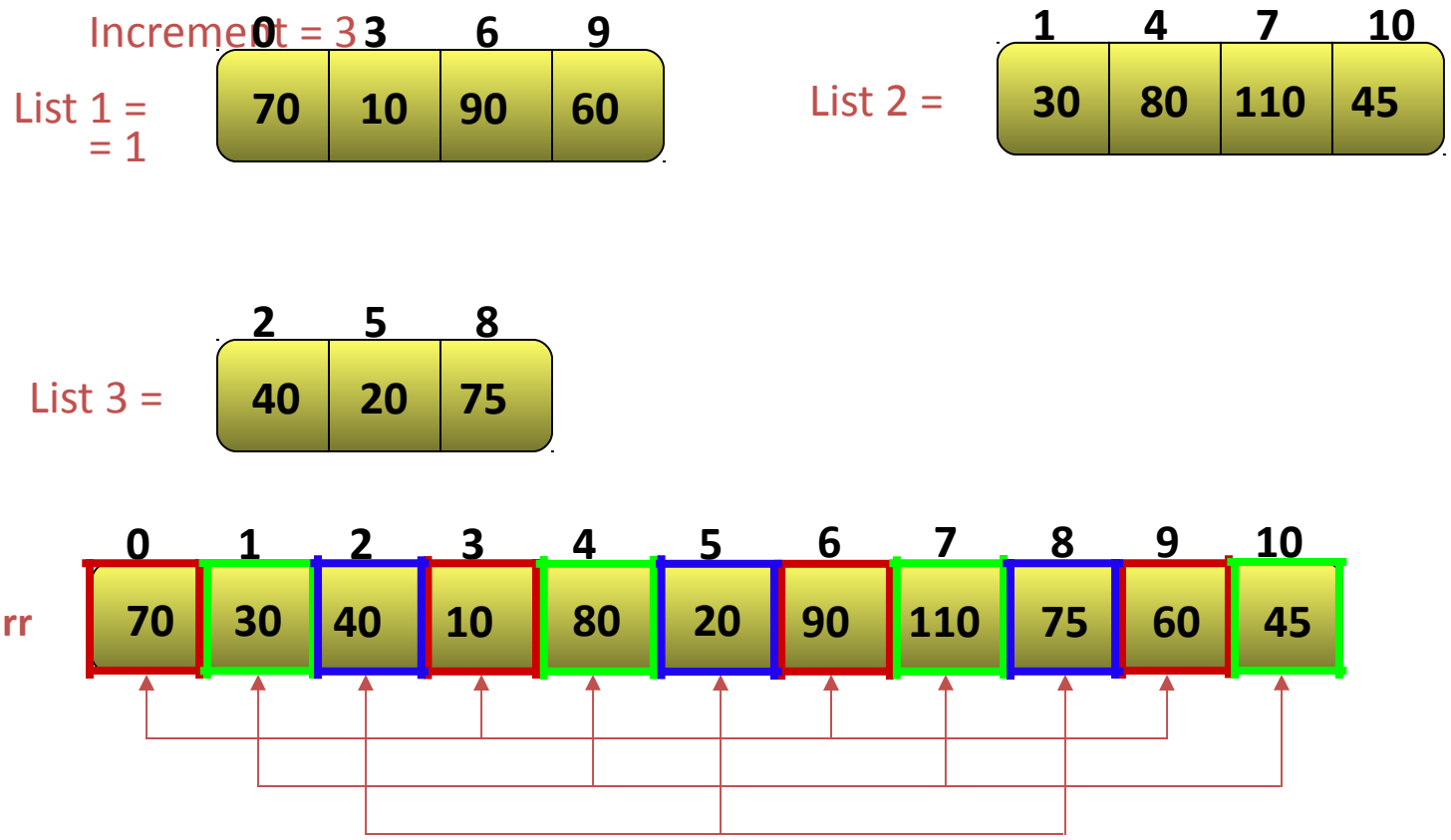


Implementing Shell Sort Algorithm (Contd.)

- To apply shell sort on this array, you need to:
  - Select the distance by which the elements in a group will be separated to form multiple sublists.
  - Apply insertion sort on each sublist to move the elements towards their correct positions.

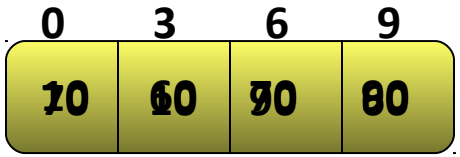


Implementing Shell Sort Algorithm (Contd.)

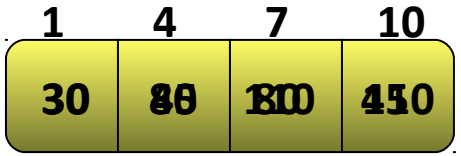


Implementing Shell Sort Algorithm (Contd.)

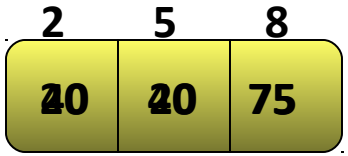
List 1 =



List 2 =



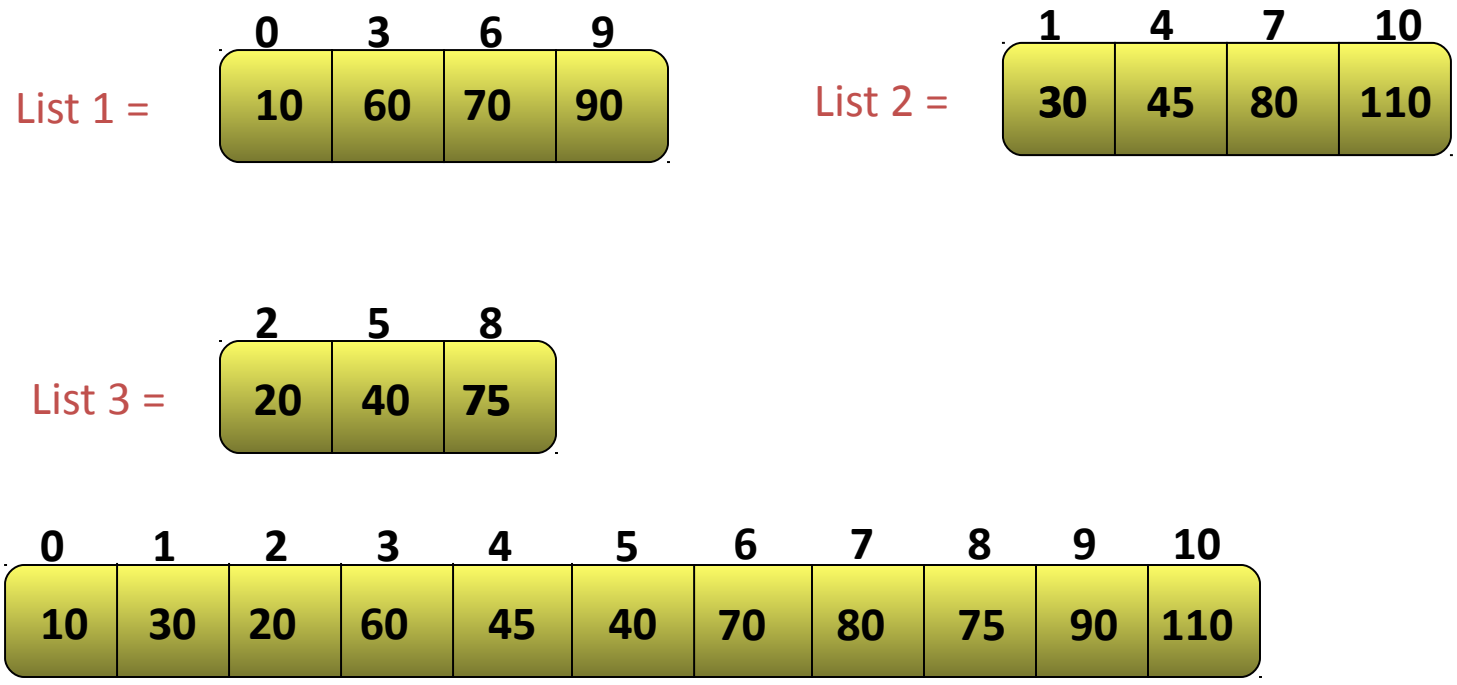
List 3 =



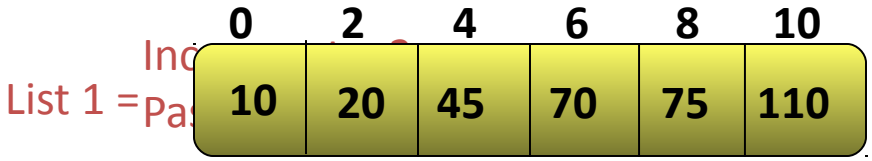
Apply insertion sort to sort the three lists

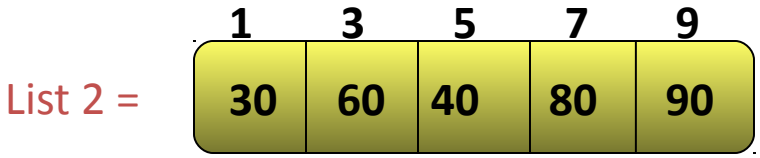
The lists are sorted

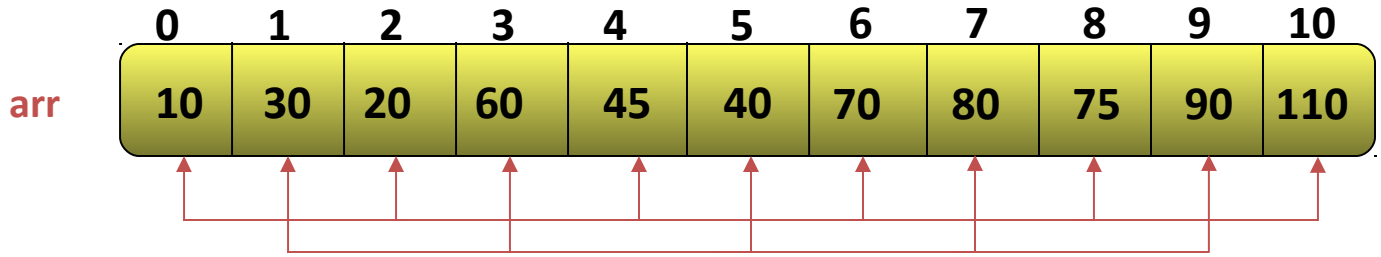
Implementing Shell Sort Algorithm (Contd.)



Implementing Shell Sort Algorithm (Contd.)

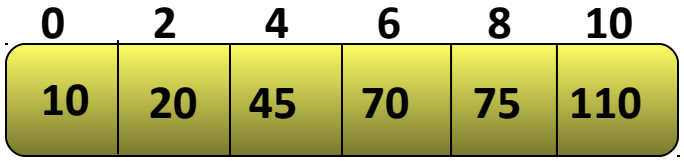




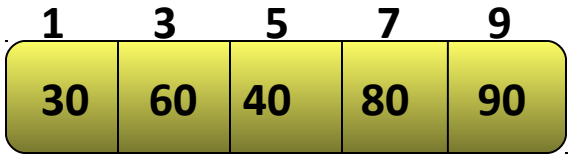


Implementing Shell Sort Algorithm (Contd.)

List 1 =



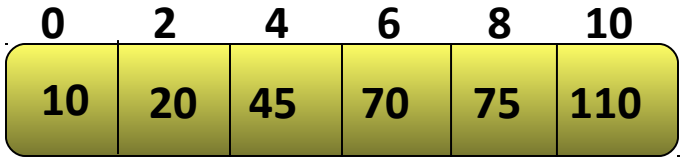
List 2 =



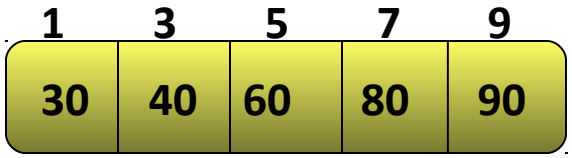
Apply insertion sort on each sublist

Implementing Shell Sort Algorithm (Contd.)

List 1 =

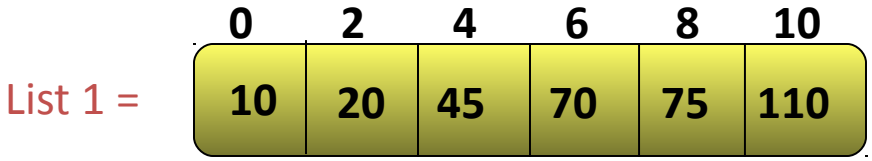


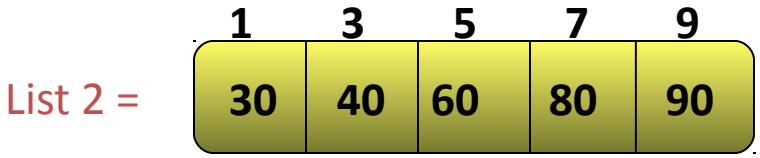
List 2 =

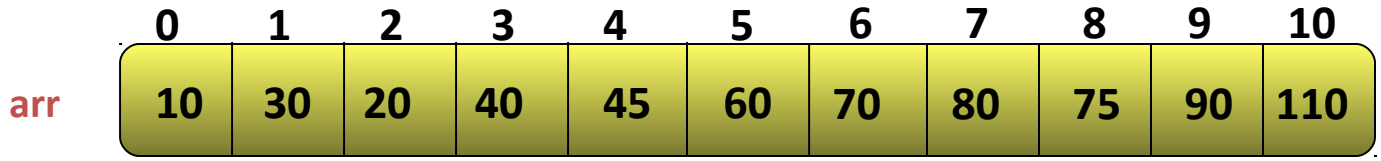


The lists are now sorted

Implementing Shell Sort Algorithm (Contd.)



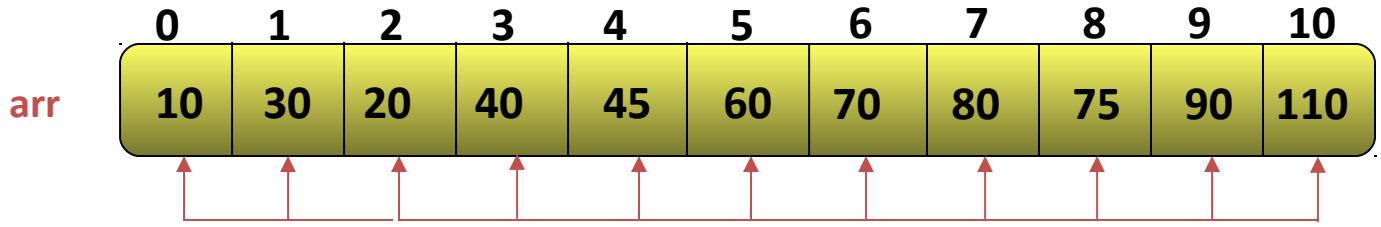




Implementing Shell Sort Algorithm (Contd.)

Increment = 1

Pass = 3

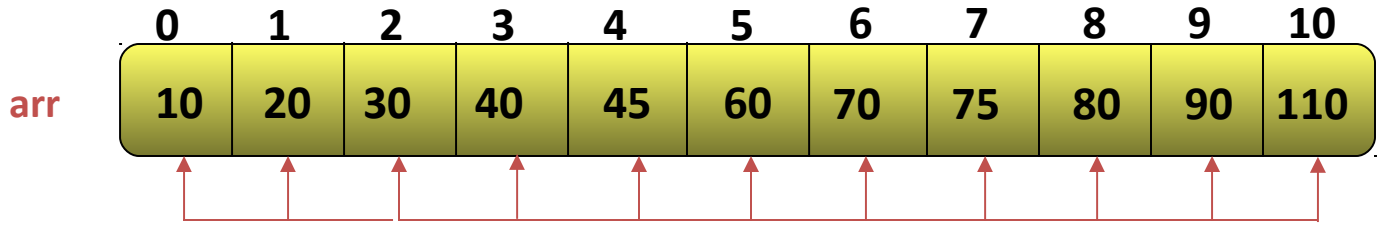


Apply insertion sort to sort the list

Implementing Shell Sort Algorithm (Contd.)

Increment = 1

Pass = 3



The list is now sorted

Algorithm shellsort (a [], n)

// a is array of n elements

1. for(gap=n/2 ; gap>0 ; gap=gap/2)

    //gap is increment

  1.1 for (i=gap ; i<n ; i ++)

  1.1.1 set temp= a [i]

  1.1.2 for (j=i; j>=gap && a[j-gap]>temp; j=j-gap)

    1.1.2.1 a[j] = a[j-gap]

  1.1.3 set a [j]=temp

Just a minute

- Which of the following sorting algorithms compares the elements separated by a distance of several positions to sort the data? The options are:
  1. Insertion sort
  2. Selection sort
  3. Bubble sort
  4. Shell sort

- Answer:
  4. Shell sort