

# Shell Sort in R

Rahul Goswami

2022-01-18

Shell sort is a sorting algorithm , usually in bubble sort we swap the adjacent elements if they are in wrong order. Shell sort is an improvement over bubble sort where it swap the elements of the array in a gap. The idea is to start with a large gap and then reduce the gap. The basic idea of shell sort is to maintain a sorted sublist in the lower positions of the list, and then pick the element from the upper part and put it in the correct position.

Lets create a function that implements the shell sort algorithm.

```
shell.sort <- function(vec){
  # Iterate over the vector
  for(i in 1:length(vec)){
    # Initialize the gap
    gap <- floor(length(vec)/2)
    # Iterate over the unsorted list
    while(gap > 0){
      # Iterate over the sublist
      for(j in (gap+1):length(vec)){
        # If the element at j is smaller than the element at j-gap
        if(vec[j] < vec[j-gap]){
          # Swap the elements
          temp <- vec[j]
          vec[j] <- vec[j-gap]
          vec[j-gap] <- temp
        }
      }
      # Decrement the gap
      gap <- floor(gap/2)
    }
  }
  vec
}
```

## Example

```
vector <- c(7,6,5,1,0,9,5,5)
shell.sort(vector)
```

```
## [1] 0 1 5 5 5 6 7 9
```