Problem5_BubbleSort

Akira MATSUI

March 27, 2017

Introduction

In this section, we will learn "Bubble Sort" which is the one of the famous algorithms. Here is the simple explanation.

There is a list that has some number.

`(2,7,1,6)`

What you need to do is to put these number in numerical order. Bubble sort is a method to do that by comparing each pair of adjacent items and swapping them if they are in the wrong order.

First Step

`(2,1,7,6)`

Second Step

`(1,2,6,7)`

On this example, the list is ordered in numerical order in second step. This algorithms is terminated when we do no change any number in that step.

Question

- you have `BS <- as.integer( runif(100, min = 1, max = 99) )`
- Imprement `bubble sort` make `BS` to be sorted

Sample Answer

```
bubble<- function(BS){
    N <- length(BS) #Number of factor
    flag = TRUE
    while(flag){
      flag = FALSE
      for (i in 2:N-1){
        if(BS[i] > BS[i+1]){
          B = BS[i]
          S = BS[i+1]
          BS[i] = S
          BS[i+1] = B
          flag = TRUE
        }
       }
```

```
    }
    return(BS)
}
BS <- as.integer( runif(100, min = 1, max = 99) )
print("##########################BEFORE##########################################")
```

```
## [1] "##########################BEFORE##########################################"
```

```
print(BS)
```

```
##   [1] 78 81 34 26 92 32 93 29 19 84 52 20 76 34 66 53 66 66 64 42 63 61 72
##  [24] 91 66 73 25 19 17 52 55 49  6 98 18 54 67 87 81 43 78 62 63 66 77  6
##  [47] 15 67 24 86 27 18 21 94  4 63 66 56 96 55 73 34 10 77 89 83 18 74 77
##  [70] 73 95 28 23  1 68 61  8 79 51 18 91  8  3 81 66 41 23 70 31  2 84 31
##  [93] 37 21 54 54 19 88 66 34
```

```
print("##########################AFTER###########################################")
```

```
## [1] "##########################AFTER###########################################"
```

```
bubble(BS)
```

```
##   [1]  1  2  3  4  6  6  8  8 10 15 17 18 18 18 18 19 19 19 20 21 21 23 23
##  [24] 24 25 26 27 28 29 31 31 32 34 34 34 34 37 41 42 43 49 51 52 52 53 54
##  [47] 54 54 55 55 56 61 61 62 63 63 63 64 66 66 66 66 66 66 66 66 67 67 68
##  [70] 70 72 73 73 73 74 76 77 77 77 78 78 79 81 81 81 83 84 84 86 87 88 89
##  [93] 91 91 92 93 94 95 96 98
```

swap function

If you want to use `swap()` function, "seqinr" library provide it.

```
install.packages("seqinr")
```

```
bubble<- function(BS){
    library(seqinr)
    N <- length(BS) #Number of factor
    flag = TRUE
    while(flag){
      flag = FALSE
      for (i in 2:N-1){
        if(BS[i] > BS[i+1]){
          swap(BS[i],BS[i+1])
          flag = TRUE
        }
      }
    }
    return(BS)
}
BS <- as.integer( runif(100, min = 1, max = 99) )
print("##########################BEFORE##########################################")
```

```
## [1] "##########################BEFORE##########################################"
```

```
print(BS)
```

```
##   [1] 73 65 22 80 65 71 47 97  5 40 27 15 38 17 98 17 51 72 91 88 72 33 56
```

```
##  [24] 87 76 29 28 36 33 82 15 71 38 94 84 62 33 87 11 33 25 52 95 15 84  5
##  [47] 92 67 69 40 38 57 96 70 73  5  1 59 34 81 16  2 80 76 13 82 66 82 37
##  [70] 34 85 85 51  4 50 25 12 13 91  4 45 79 61 81 44 10 26 54 75 65 73 13
##  [93] 10 76 50 35 26 19 58  1
```

```
print("##########################AFTER##########################################")
```

```
## [1] "##########################AFTER##########################################"
```

```
bubble(BS)
```

```
##   [1]  1  1  2  4  4  5  5  5 10 10 11 12 13 13 13 15 15 15 16 17 17 19 22
##  [24] 25 25 26 26 27 28 29 33 33 33 33 34 34 35 36 37 38 38 38 40 40 44 45
##  [47] 47 50 50 51 51 52 54 56 57 58 59 61 62 65 65 65 66 67 69 70 71 71 72
##  [70] 72 73 73 73 75 76 76 76 79 80 80 81 81 82 82 82 84 84 85 85 87 87 88
##  [93] 91 91 92 94 95 96 97 98
```