

Problem6: BinarySearch

Akira MATSUI

March 27, 2017

Introduction

In this exercise, we will learn binary search algorithm, which is to find a specific number in a sorted list. To use this search algorithm, you have to make sure that your list is ordered numerically. This exercise is related to bubble sort exercise.

If you have a number A that you need to find out where it is in a list, binary search algorithm tell you that in following steps. Suppose the list has N elements.

- compare A with the mean number R_m , of the list ($m := N/2$)
- if $A > R_m$, compare A with R_{m_2} ($m_2 := 3N/4$)
- if $A < R_m$, compare A with R_{m_2} ($m_2 := N/4$)

Here is the simple explanation.

There is a list that has numbers. Suppose that you want to find where $T = 9$ is, but cannot find it at a glance. This might happens if you list has a hundreds of elements.

(1,2,3,6,7,8,9)

First Step

Compare T with 6. In this case, $T > 6$ then go to next step.

Second Step

Divide the list into half.

(1,2,3), (7,8,9)

Because $T > 6$, we are interested in the latter list (7,8,9).

Compare T with 8. In this case, $T > 8$ then we have found where $T = 9$ is in the list.

Question

- you have `BS <- sort(as.integer(runif(100, min = 1, max = 99)))`
- Implement `binary search` find where T is in `BS`
- In this excersise, set `T <- as.integer(runif(1, min = 1, max = 99).`
- In the case `BS` dose not have the same number as `T`, `print('can't find')`

Sample Answer