Jump Search in R

Rahul Goswami

2022-01-18

Jump Search is a searching algorithm that is used to find an element in a sorted array. The basic idea of jump search is to search for an element in a sorted array by jumping ahead by fixed steps. The average case time complexity of jump search is O(sqrt(n))
The worst case time complexity of jump search is O(n)
The best case time complexity of jump search is O(sqrt(n)) (if the array is already sorted)
The space complexity of jump search is O(1) (in-place)

@param vec Vector to be searched
@param element Element to be searched
@return Index of the element

```
jump.search <- function(vec, element){
    jump_step <- as.integer(sqrt(length(vec)))
    prev_index <- 1
    while(vec[prev_index] < element){
        prev_index <- prev_index + jump_step
        if(prev_index >= length(vec)){
            return("Not Found")
        }
    }

    while(vec[prev_index] < element){
        prev_index <- prev_index + 1

    }

    prev_index
}
```

Example

```
sorted_vec <- c(0,1,2,3,4,5,6,7,8,9)
jump.search(sorted_vec, 5)
```

```
## [1] 7
```