

CS 570: Analysis of Algorithms – H5

Submitted by: Indronil Bhattacharjee

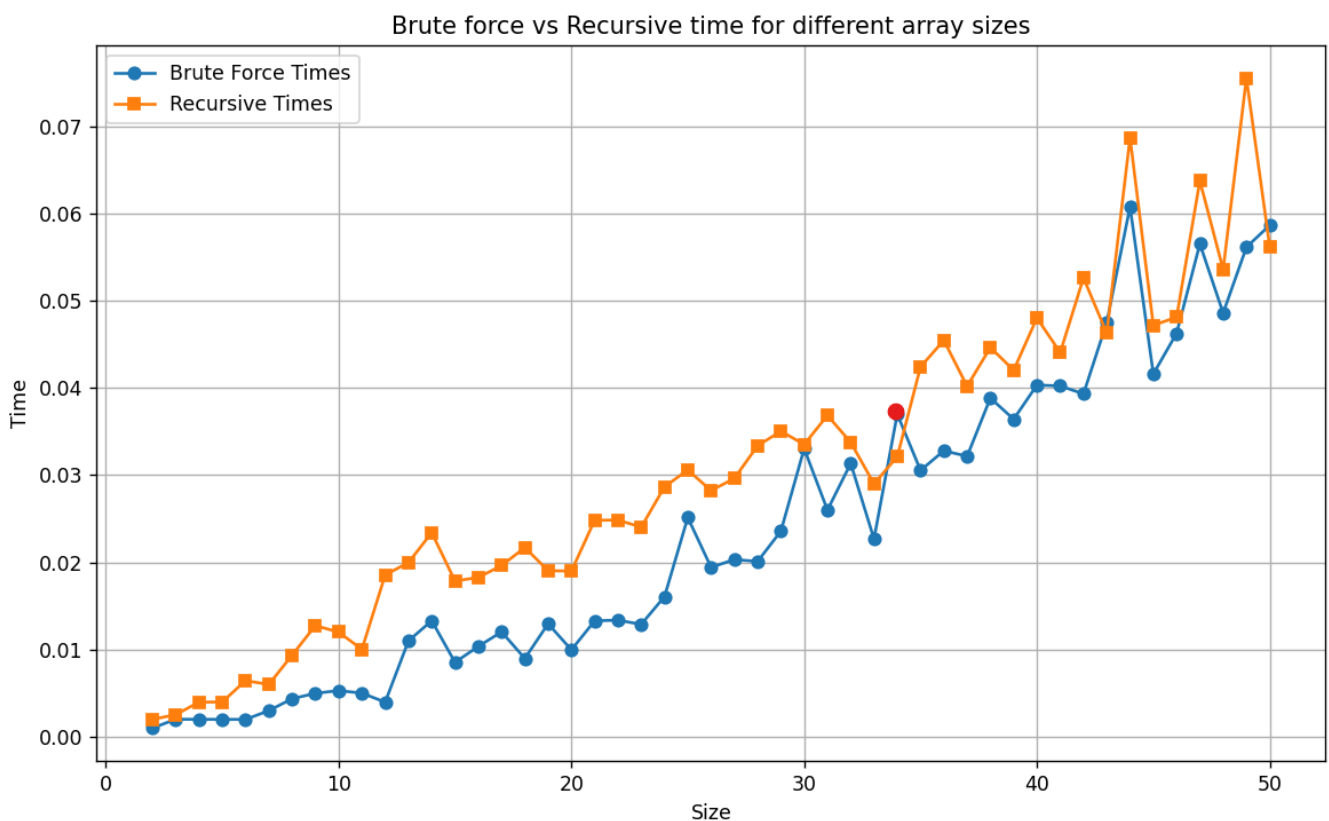
Exercise 4.1-2

FindMaxSubarray(A, low, high):

```
    left = 0
    right = 0
    sum =  $-\infty$ 
    for i = low to high
        current_sum = 0
        for j = i to high
            current_sum += A[j]
            if sum < current_sum
                sum = current_sum
        left = i
        right = j
    return (left, right, sum)
```

Exercise 4.1-3

In my computer, the value of n_0 is 34. You can have a look into the times required for brute force and recursive approach for array sizes starting from 2 to 50 in the following plot.



If we modify the base case of the recursive algorithm to use the brute-force algorithm whenever the problem size is less than n_0 , the crossover point does not change significantly.

However, in most cases, I have noticed there is not a single crossover point. Here, recursive beats brute-force at $n=34$, but after that, at $n=43$, brute-force beats recursive and then recursive beats brute-force again for $n=44$ to 49 , again recursive beats brute-force at $n=50$.

(The python code file MaxSubarray.py is also submitted with the assignment)