

Data Structures and Algorithms

(CO 322)

Lab 1

Name : Jayathilaka H.A.D.T.T.

E.Number : E/16/156

Date: 09/07/2020

Submit an essay which answers the following questions:

(a) Is there a difference in the runtime between the two implementations when the problem is small?

(b) Is there a difference in the runtime between the two languages?

(c) Is there a difference between the way the runtime changes in the two languages?

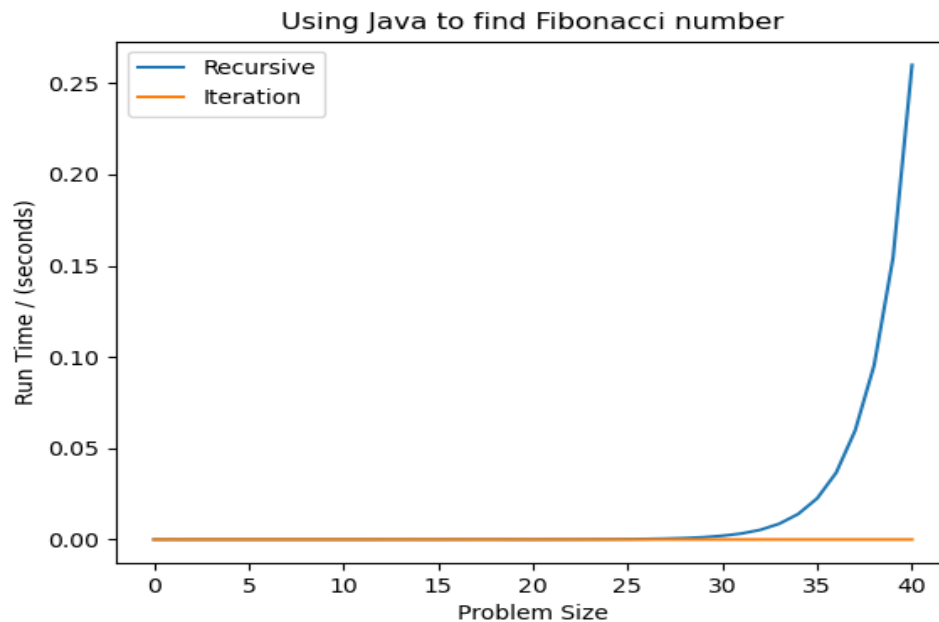
(d) "If the problem is small both algorithms are useful". Do you agree with this statement? Justify your answer.

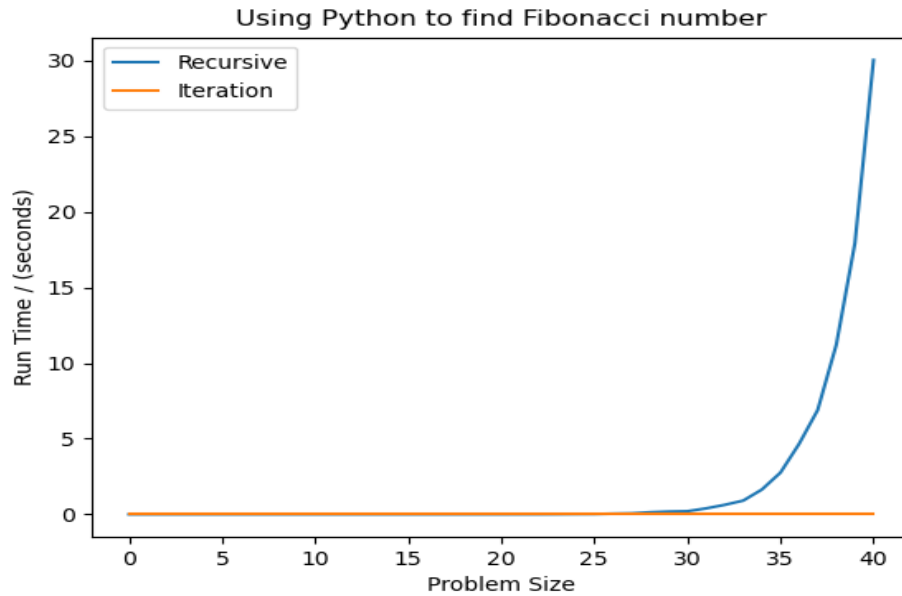
(e) "If the problem is large fib r is not useful". Do you agree with this statement? Justify your answer.

The running time of a specific input depends on the number of operations executed. According to the given input or the problem size, the number of operations that execute can be increased or decreased and it affects the running time. In this lab we are given to calculate the given Fibonacci number using two different implementation and two different programming languages. The input can differ from 1 to 40.

In order to find most suitable implementation for this algorithm we use run time as a helping key. How the run time vary according to the problem size can be easily got to a graph and using that we can compare the things in each graph. Through the similarities and the differences of the graphs we can get a better idea about the run time and how it vary according to the implementation, problem size and programming language.

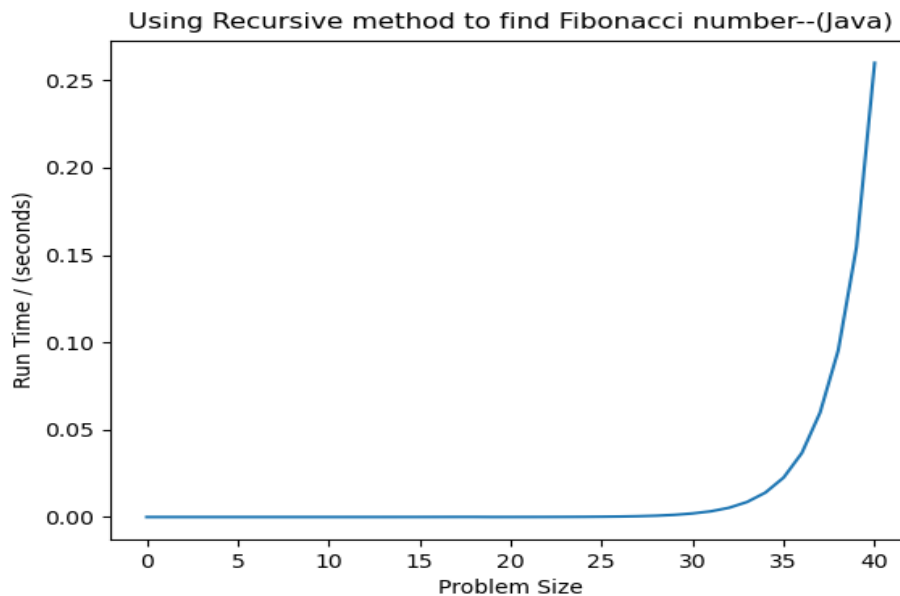
When using java and python, for both implementations (recursive and iteration) there is not a huge run time difference when the problem is small. It only take few milliseconds or nanoseconds to complete the given task.

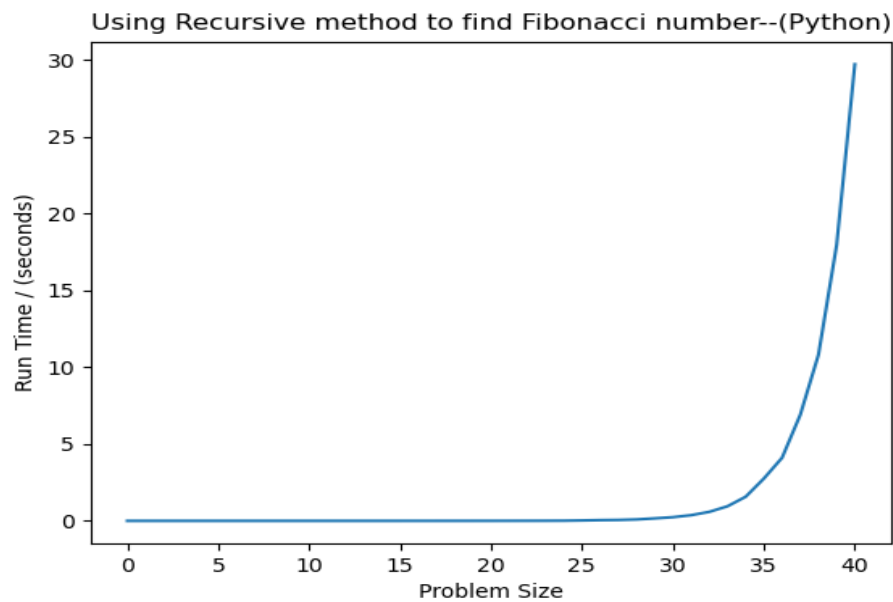




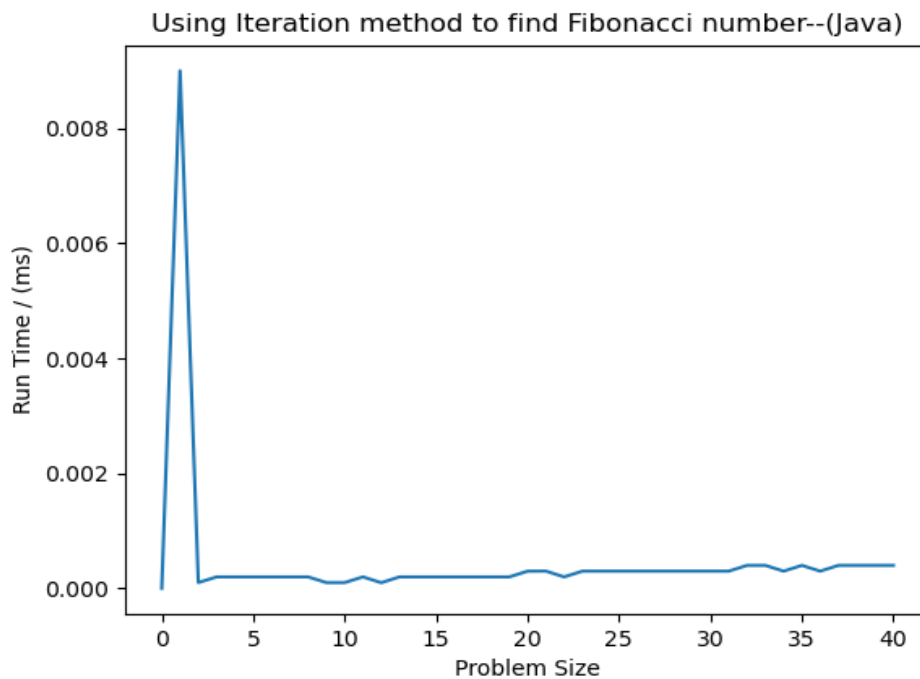
Above graphs show the run time of two implementations using both languages. In there we can clearly see how run time does not change with the problem size, when the problem size is smaller than 30. So when the problem is small, we can use any implementation which is easy to code.

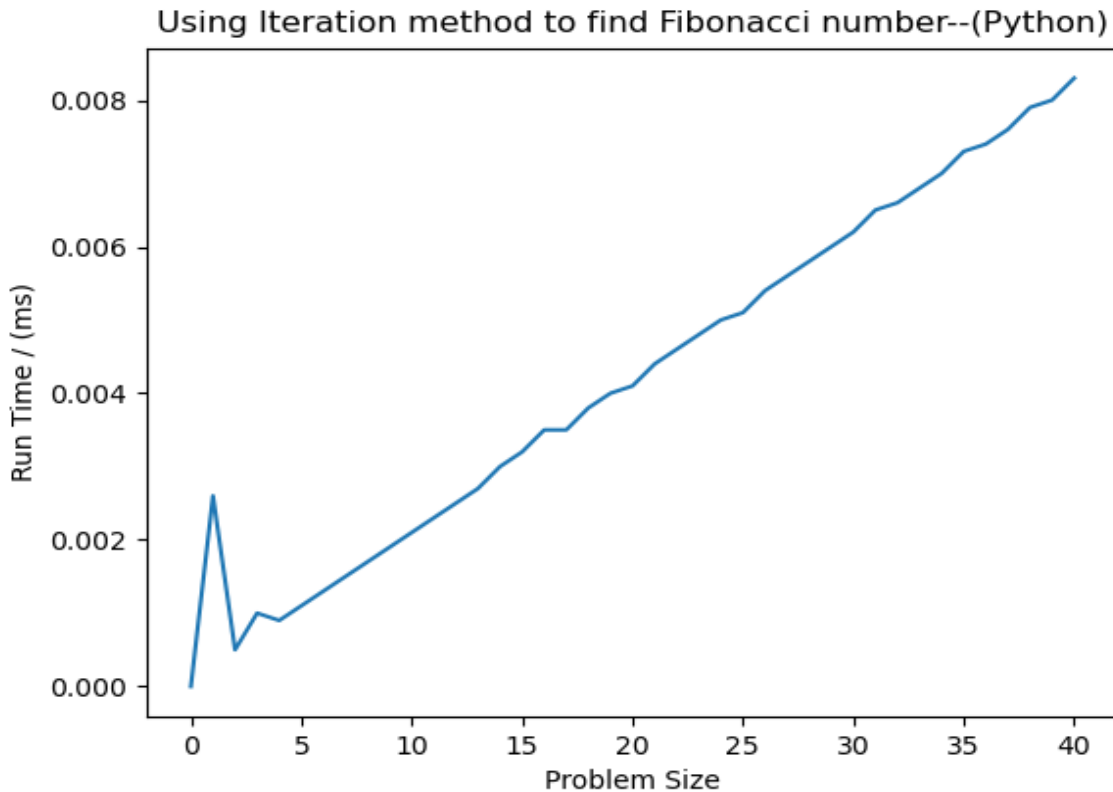
When we consider the running time of two coding languages which we used to implement the above algorithm, a huge time difference can be seen clearly.





Let us take an example. Java programming language take about 0.25 seconds to run the program when x is equal to 40, while Python programming language take about 30 second to run the same method when x is equal to 40.

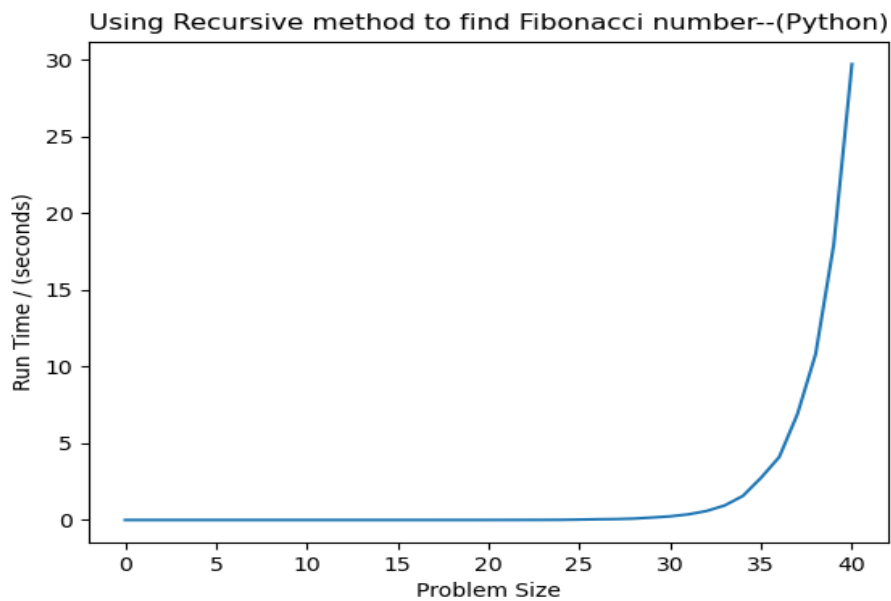
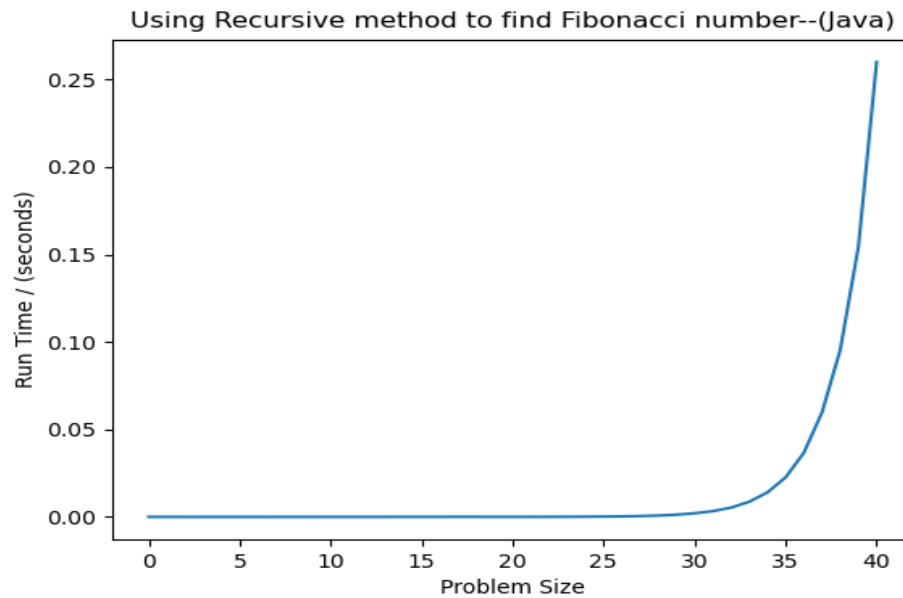




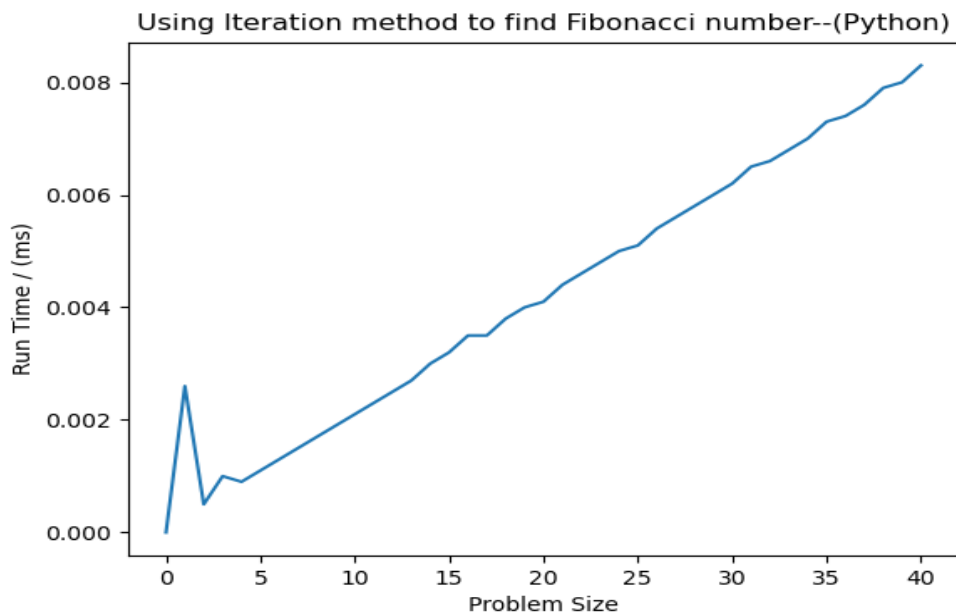
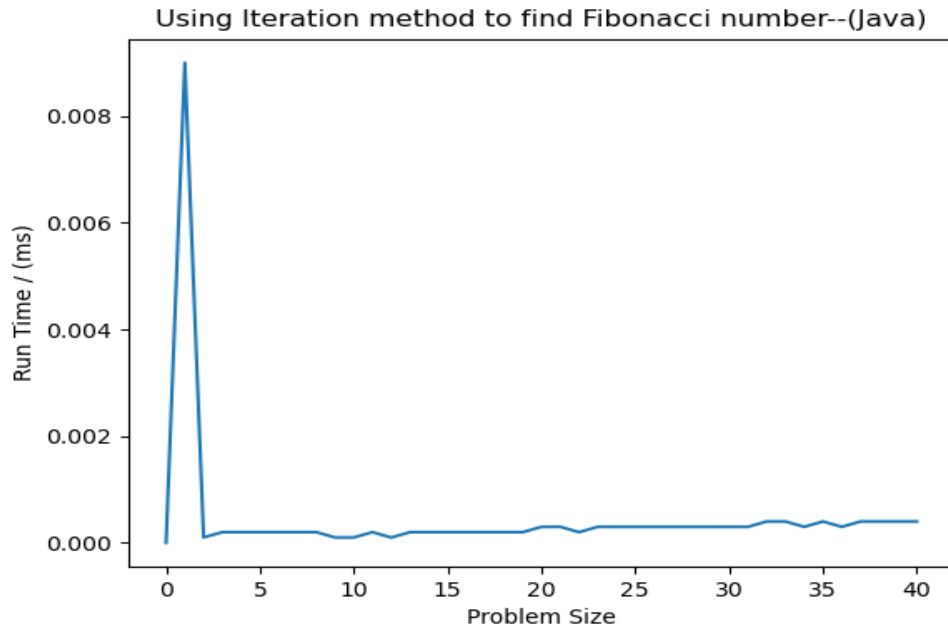
When we consider the iteration method also a huge time difference between two programming languages can be seen. For an example, Java takes about 0.0001ms to run the code when x is 40, while python takes more than 0.008ms to run the code when x is 40.

So when the problem is bigger it is better to use programming language like Java to reduce run time and get the answer quickly.

Let us consider the behaviour of the graphs according to the programming language. If the graphs show the same behavior, that means the way that run time change in both languages are same.



Recursive method is used using two different programming languages in above two graphs. We can clearly see the behaviour of above two graph is same although there is a huge difference between the run time.

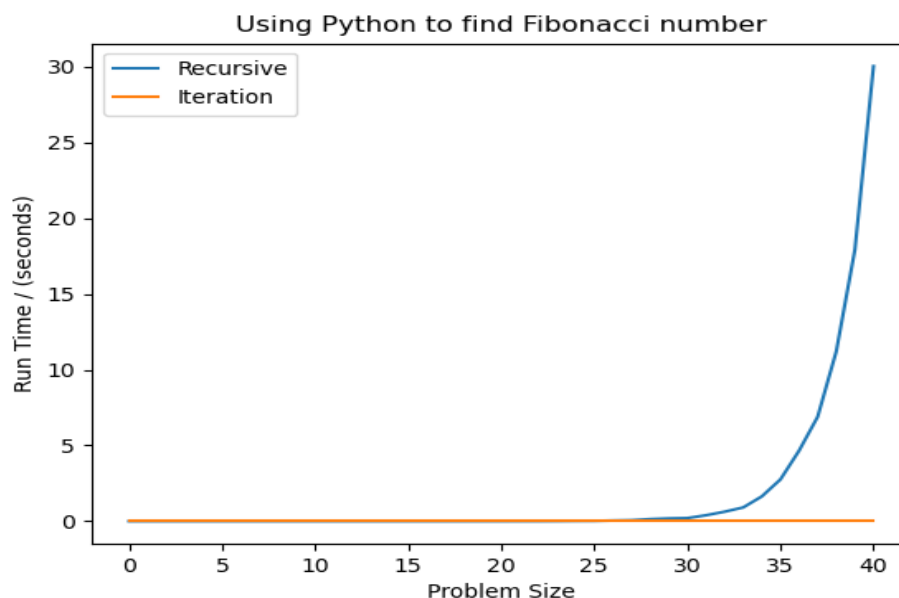
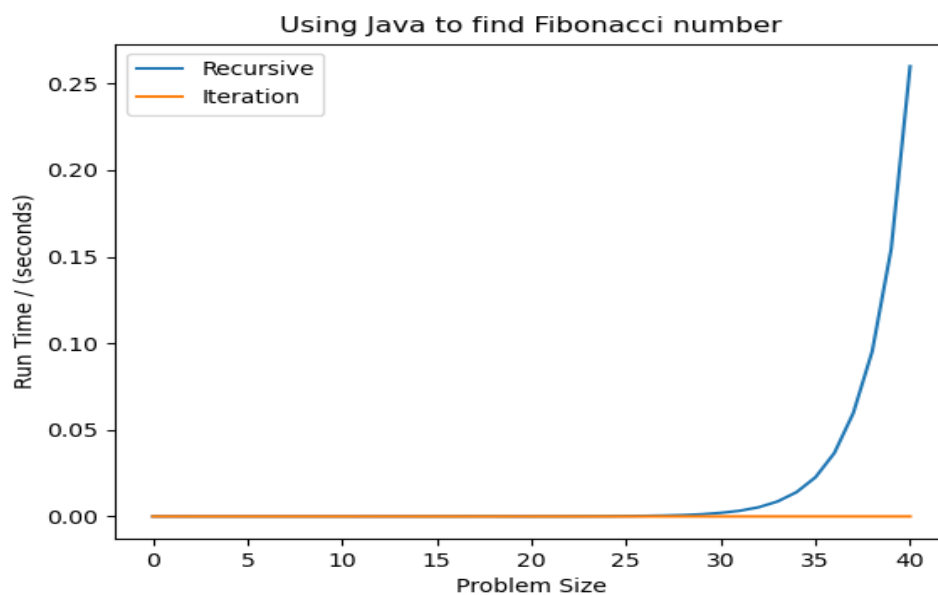


Although there are slight changes between the two graphs, we can see overall both graphs are the same. When x is equal to 1 it increased suddenly and from

2 onward it increase gradually. In first graph the the increasement cannot see clearly, but there is a small incrasement. So the behaviors of both graphs are same.

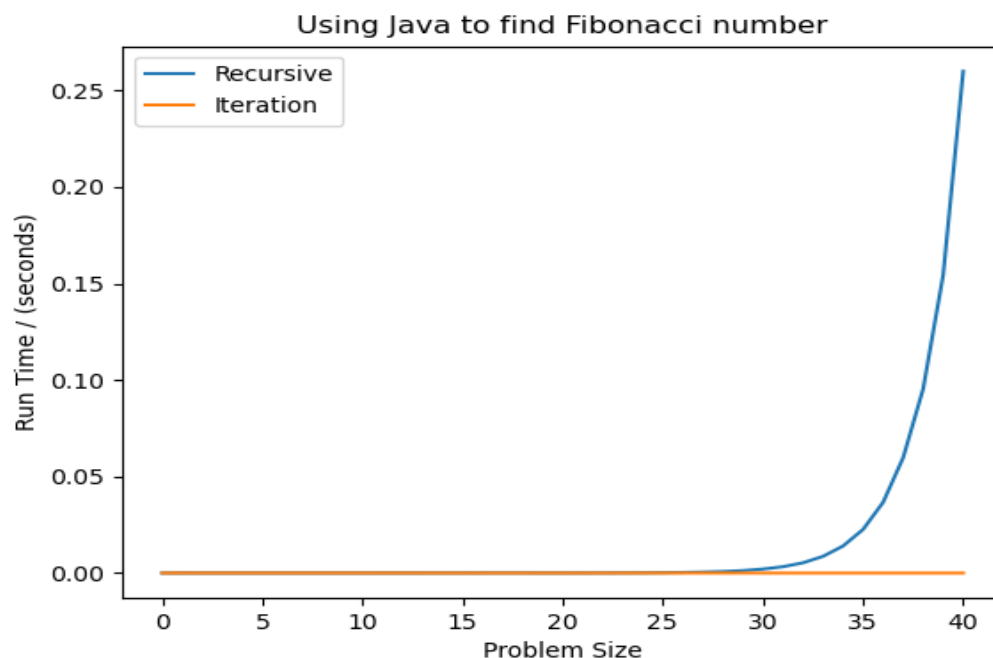
By the above facts we can see the way run time behave, is same with both languages and the method that use for implementation does matter the behaviour not the programming language that has been used.

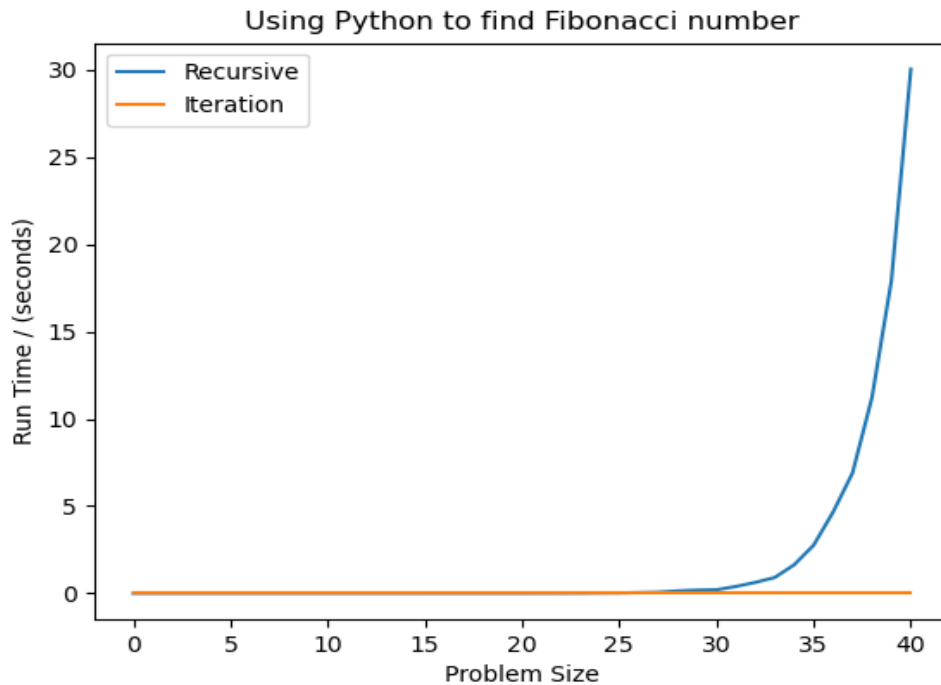
“If the problem is small both algorithms are useful”. This statement can be taken as a true. Let us consider the graphs in order to get a better idea.



In the above graphs, when the problem is small run time does not change at all, because the time taken to run those codes is few milliseconds or few nanoseconds in both algorithms and in both programming languages. Practically few milliseconds or few nanoseconds do not matter us like seconds. So the time taken to run the codes, when the problem is small can be taken as same in both algorithms. By that we can prove that the statement “If the problem is small both algorithms are useful” is true. Not only that we can also prove both programming languages are also useful when the problem is small by that.

“If the problem is large fib_r is not useful” This statement is also true. The time taken to recursion can be found by finding the value of n^{th} recursive call in terms of the previous calls. So that, finding the destination case in terms of the base case, and solving in terms of the base case gives us an idea of the time complexity of recursive equations. So when comparing recursive and iteration method, Recursive method is not useful for large problems, because it is so time consuming. We can see that in the below graphs also.





In both programming languages when the problem is high, recursive method is not appropriate. A huge change in the run time can be seen when the problem is large in both graphs. When x is equal to 40, java code takes about 0.26 seconds and python code takes about 30 seconds. If we run the same problem using the same x value using iteration method, both java code and python code take only few milliseconds or nanoseconds. So to have results quickly, we cannot use recursive method.

Hence by using above graphs also, “If the problem is large fib_r is not useful” statement can be proven true.