Calculating Execution Time of a Python Program

The execution or running time of the program indicates how quickly the output is delivered based on the algorithm you used to solve the problem. To calculate the execution time of the program, we need to calculate the time taken by the program from its initiation to the final result. So if you want to learn how to calculate the execution time of a Python program, this article is for you. In this article, I will take you through a tutorial on calculating the execution time of a Python program.

In [ ]:

Worst Case, Average Case, and Best Case

Introduction to Worst Case, Average Case and Best Case

1. Worst Case Analysis:

In the worst-case analysis, we calculate the upper limit of the execution time of an algorithm. It is necessary to know the case which causes the execution of the maximum number of operations.

For linear search, the worst case occurs when the element to search for is not present in the array. When x is not present, the search () function compares it with all the elements of arr [] one by one. Therefore, the temporal complexity of the worst case of linear search would be Θ (n).

2. Average Case Analysis:

In the average case analysis, we take all possible inputs and calculate the computation time for all inputs. Add up all the calculated values and divide the sum by the total number of entries.

We need to predict the distribution of cases. For the linear search problem, assume that all cases are uniformly distributed. So we add all the cases and divide the sum by (n + 1).

3. Best Case Analysis:

In the best case analysis, we calculate the lower bound of the execution time of an algorithm. It is necessary to know the case which causes the execution of the minimum number of operations. In the linear search problem, the best case occurs when x is present at the first location.

The number of operations in the best case is constant. The best-case time complexity would therefore be Θ (1) Most of the time, we perform worst-case analysis to analyze algorithms. In the worst analysis, we guarantee an upper bound on the execution time of an algorithm which is good information.

In [ ]:

So to calculate the execution time of a Python program, we need to follow the steps mentioned below:

1. First, store the time of initiation of the program into a variable;
2. Write the Python program;
3. Store the end time of the program into a variable;
4. Subtract the time of initiation of the program from the end time of the program;

In [ ]:

Calculating Execution Time of a Python Program

Now let's follow the process described in the above section to calculate the time taken by a Python program. Here I will write a simple program to create acronyms:

In [2]:
```
from time import time
start = time()

# Python program to create acronyms
word = "Artificial Intelligence"
text = word.split()
a = " "
for i in text:
    a = a+str(i[0]).upper()
print(a)

end = time()
execution_time = end - start
print("Execution Time : ", execution_time)
```

```
 AI
Execution Time :  0.0002675056457519531
```

In [ ]:

Summary

So this is how you can find the execution time of your program. For calculating the execution time of a program, we calculate the time taken by the program from its initiation till the final

output. I hope you liked this article on calculating the running time of a Python program. Feel free to ask valuable questions in the comments section below.

In [ ]: