Big O Notation



Graphical user interface, application

Description automatically generated

What is Big O notation?  
Big O notation is used to analyze the efficiency of an algorithm as its input approaches infinity.

As the size of the INPUT to the algorithm grows, how drastically does the space or time requirements grow with it.

O(n) – suggests that an operation takes the same time per item (n) to process regardless of how many items (n) there are.

Asymptotic

O(n)2

|  |  |
| --- | --- |
|  |  |
| A picture containing graphical user interface  Description automatically generated | Time required to process is approximately the square of the amount of records / bytes / bits / etc.  “… the processing time required by the Algorithm will be the square of the data input.”  Time = O | ( records = n ) 2 |

**Big O(n) Function sample**

Graphical user interface, text

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Ignore the Constants

What is a Constant?

A Constant is any step that doesn’t scale with the INPUT to the function. For example, the time to evaluate this expression does not change with the INPUT. Below: 100 and 1000 are Constants.

**Big O (1) – Constants / Constants Algorithm Function Sample**

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated with medium confidence

Growth Hierarchy

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Source: <https://www.bigocheatsheet.com/>

Source: <https://www.youtube.com/watch?v=V6mKVRU1evU&list=PLGLfVvz_LVvReUrWr94U-ZMgjYTQ538nT>