Lecture-04-CMSC351

Rigorous Time:

- In the simplest case, the time complexity of code depends on some *n*, which could be the length of a list or the number of times a loop iterates, etc
- Our goal is to imagine a function, T(n), that could tell us how much time the code takes for any *n*, and then find a simple *n* such that T(n) = theta(n) when possible
- Assignments (var = value) take constant time
- The time complexity of a loop can be reduced to the highest degree function
- Assignments, for loops, while loops, and conditionals all take time to perform
- In a worst-case scenario a conditional is assumed to be true and the entire conditional can be replaced by the time the body takes
- Able to ignore maintenance line when evaluating time complexity of loops, only evaluate loop body instead