

# CSC510 Lecture 1

Chris Camano: ccamano@sfsu.edu

August 23, 2022

## Algorithm definition and Algorithm Analysis philosophy

Algorithms are generally described using pseudocode as an abstract expression of a procedure. It is independent from any programming language and follows the convention of mathematics instead.

We will focus on proving that a given algorithm always produces the right answer and we will also describe its worst case time complexity.

Formally showing correctness of recursive algorithms uses proof by induction.

## Time complexity of common algorithms

Merge sort:  $O(n \log(n))$

Insertion sort:  $O(n^2)$  **Recursion/ Divide and conquer**

Many problems have a recursive structure. Big solution from smaller solutions. **Backtracking**

An incremental brute force approach in which a recursive algorithm is run until a solution criteria is satisfied.

## Dynamic Programming

$$F_1 = 1 \quad F_2 = 1$$

$$F_n = F_{n-1} + F_{n-2} \quad \forall n \in \mathbb{Z}, n > 2$$

```
Fib(n
if(n=1 or n=2):
return 1;
else :
Fib(n-1)+Fib(n-2)
```

This algorithm is very inefficient because similar components are being recomputed.

instead of computing from the top to bottom it actually can be much more efficient to save intermediate solutions to avoid redundant work.

```
DpFib(n)
init F[0,...,max(1,n)]
F[0]:=0
F[1]:=1
for i=2,...,n
F[i]:=F[i-1]+F[i-2]
```

return F[n]

## Lecture 1

### Big O notation

Big O notation allows us to isolate bottlenecks in algorithms independent of a implementation. It also ignores discrepancies between systems and captures long term behavior.

Big O notation is asymptotic notation, it has to do with a formal definition centered on the concept of a limit.

$O(f(n))$ : class of functions whose order is at most  $f(n)$ .

A big O classification is a family of function.

$$f(n) \in O(g(n))$$

$f(n)$  is an element of the set of functions bounded by  $g(n)$ . In computer science this set inclusion operator is defined as equivilancy.

Let  $T(n)$  be the of operations when running an algorithm on size  $n$  input.

$$\exists_{C,K \in \mathbb{Z}}, K > 0 : \quad T(n) \leq C f(n) \quad \forall_n > K$$