



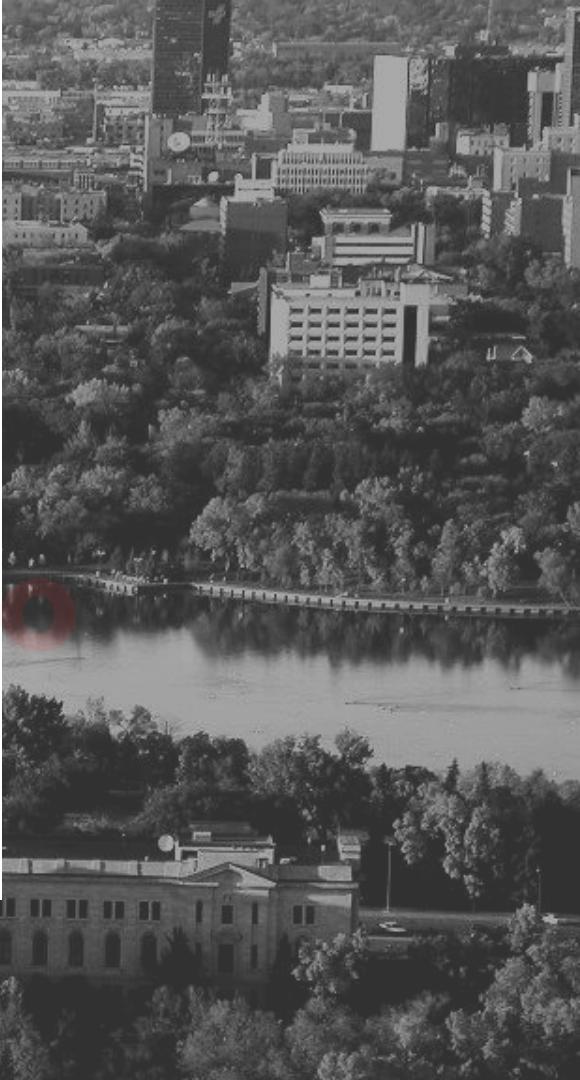
UNIVERSITY OF REGINA

CS310-002
DISCRETE
COMPUTATIONAL
STRUCTURES
andreeds.github.io

ANDRÉ E. DOS SANTOS

dossantos@cs.uregina.ca

andreeds.github.io



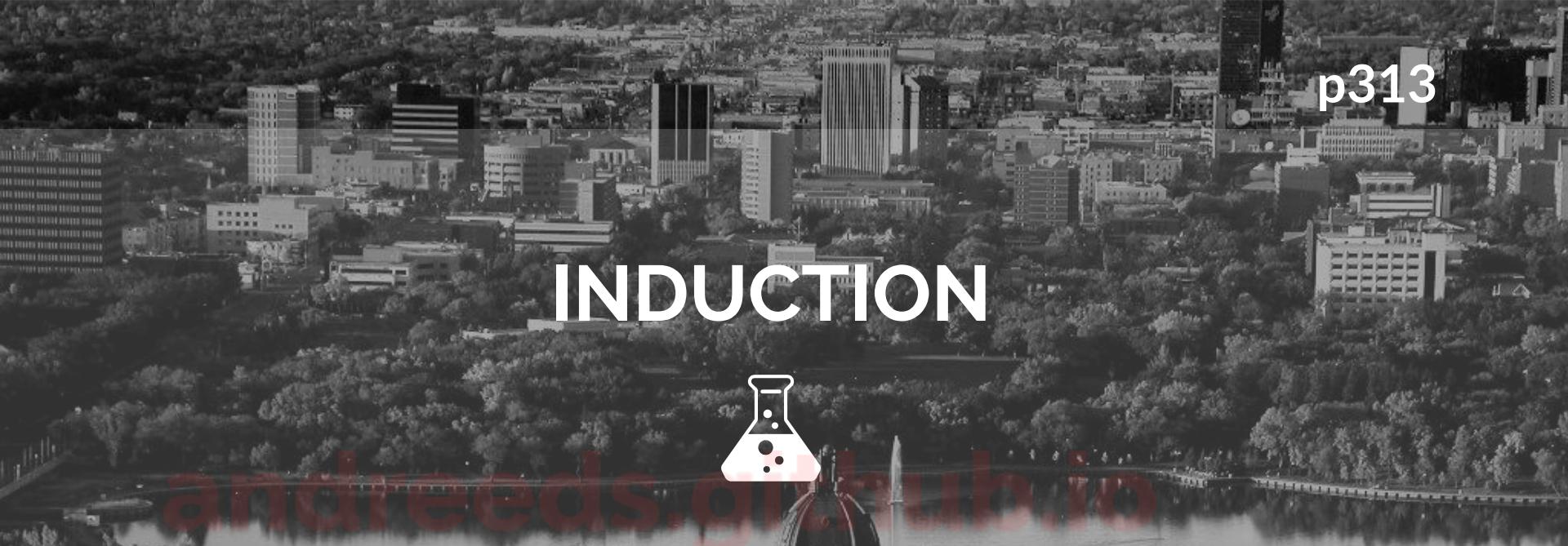
CS310-002
DISCRETE COMPUTATIONAL
STRUCTURES

INDUCTION AND RECURSION

andreeds.github.io

ANDRÉ E. DOS SANTOS
dossantos@cs.uregina.ca
andreeds.github.io



A black and white aerial photograph of a city skyline, likely Edmonton, Alberta, Canada. The city is built on a hillside overlooking a wide river. In the foreground, there's a large area of green trees and some low-rise buildings. The middle ground shows a mix of residential and commercial buildings, with several tall skyscrapers rising above the city. The background features more hills and buildings under a clear sky.

p313

INDUCTION



andrews.cs@utb.edu

INDUCTION AND RECURSION

PRINCIPLE OF MATHEMATICAL INDUCTION

To prove that $P(n)$ is true for all positive integers n , where $P(n)$ is a propositional function, we complete two steps:

i. BASIS STEP

We verify that $P(1)$ is true

ii. INDUCTIVE STEP

We show that the conditional statement

$P(k) \rightarrow P(k + 1)$ is true for all positive integers k

INDUCTION AND RECURSION

PRINCIPLE OF MATHEMATICAL INDUCTION

This proof technique can be stated as:

$$(P(1) \wedge \forall k (P(k) \rightarrow P(k + 1))) \rightarrow \forall n P(n) ,$$

when the domain is the set of positive integers

INDUCTION AND RECURSION

Example:

Use mathematical induction to show that

$$1 + 2 + \dots + n = \frac{n(n + 1)}{2}$$

andreeds.github.io

INDUCTION AND RECURSION

Example:

Use mathematical induction to show that

$$1 + 2 + 2^2 + \cdots + 2^n = 2^{n+1} - 1$$

andreeeds.github.io

p334

STRONG INDUCTION



andrews.gitbook.io

INDUCTION AND RECURSION

PRINCIPLE OF STRONG INDUCTION

To prove that $P(n)$ is true for all positive integers n , where $P(n)$ is a propositional function, we complete two steps:

i. BASIS STEP

We verify that $P(1)$ is true

ii. INDUCTIVE STEP

We show that the conditional statement

$[P(1) \wedge P(2) \wedge \dots \wedge P(k)] \rightarrow P(k + 1)$ is true for all positive integers k

INDUCTION AND RECURSION

PRINCIPLE OF STRONG INDUCTION

This proof technique can be stated as:

$$P(1) \wedge \forall k ([P(1) \wedge P(2) \wedge \dots \wedge P(k)] \rightarrow P(k + 1))) \rightarrow$$

$$\forall n P(n),$$

andreeds.github.io

when the domain is the set of positive integers

INDUCTION AND RECURSION

p336

Example:

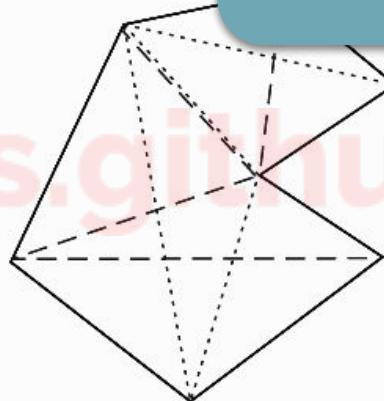
Show that if n is an integer greater than 1, then n can be written as the product of primes.

andreeds.github.io

WORTH CHECKING OUT

p339

A simple polygon with n sides, where n is an integer with $n \geq 3$, can be triangulated into $n - 2$ triangles



andreevs.github.io

RECURSIVE ALGORITHMS



INDUCTION AND RECURSION

PRINCIPLE OF MATHEMATICAL RECURSION

We use two steps to define a function with the set of nonnegative integers as its domain:

i. BASIS STEP

Specify the value of the function at zero.

ii. RECURSIVE STEP

Give a rule for finding its value at an integer from its values at smaller integers

INDUCTION AND RECURSION

Example:

Give a recursive definition of

$$\sum_{k=0}^n a_k.$$

andreeds.github.io

INDUCTION AND RECURSION

Definition

An algorithm is called **recursive** if it solves a problem by reducing it to an instance of the same problem with smaller input

Example

andreeds.github.io

Fibonacci numbers is a sequence of integers **0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...**, which can be generated by the following recurrence equation:

$$F_0 = 0, \quad F_1 = 1, \quad F_n = F_{n-1} + F_{n-2} \text{ for } n \geq 2$$

A ITERATIVE ALGORITHM FOR FIBONACCI NUMBERS

Input: A nonnegative integer n .
Output: The n -th Fibonacci number.

```
1 if  $n = 0$  then
2   return 0
3 else
4    $x \leftarrow 0$  ;
5    $y \leftarrow 1$  ;
6   for  $i = 1$  to  $n - 1$  do
7      $z \leftarrow x + y$  ;
8      $x \leftarrow y$  ;
9      $y \leftarrow z$  ;
10 return  $y$ 
```

A RECURSIVE ALGORITHM FOR FIBONACCI NUMBERS

Input: A nonnegative integer n .

Output: The n -th Fibonacci number.

```
1 Procedure fibonacci( $n$ : nonnegative integer)
2   if  $n = 0$  then
3     return 0
4   else if  $n = 1$  then
5     return 1
6   else
7     return fibonacci( $n-1$ ) + fibonacci( $n-2$ )
```

Example:

Give a recursive algorithm for computing $n!$, where n is a nonnegative integer.

andreeds.github.io

INDUCTION AND RECURSION

Example:

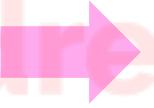
Give a recursive algorithm for computing the greatest common divisor of two nonnegative integers a and b with $a < b$.

andreeds.github.io

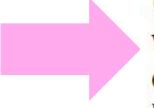


REVIEW QUESTIONS

INDUCTION AND STRONG INDUCTION

and  [github.io](https://andrews.github.io)

$$1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}$$

 Use strong induction to show that if a simple polygon with at least four sides is triangulated, then at least two of the triangles in the triangulation have two sides that border the exterior of the polygon.



REVIEW QUESTIONS

RECURSION

and

→ Describe a recursive algorithm for multiplying two non-negative integers x and y based on the fact that $xy = 2(x \cdot (y/2))$ when y is even and $xy = 2(x \cdot \lfloor y/2 \rfloor) + x$ when y is odd, together with the initial condition $xy = 0$ when $y = 0$.



→ Give iterative and recursive algorithms for finding the n th term of the sequence defined by $a_0 = 1$, $a_1 = 3$, $a_2 = 5$, and $a_n = a_{n-1} \cdot a_{n-2}^2 \cdot a_{n-3}^3$. Which is more efficient?