



CS 113 – Computer Science I

Lecture 10 - Recursion

Adam Poliak

02/16/2023

Announcements

- HW03 – due tonight night
- HW04 - releasing tonight
 - Due Wednesday 02/22
- Midterm in class Thursday 03/02
 - Closed book



Agenda

Recap

Recursion Examples

4 problems



Washing dishes

Smart way to wash dishes

Punt the problem to someone else

But we want to wash one dish so we can say we washed a dish

Recursion

a function that calls itself



“Simple” way to solve “similar” problems

Creating a recursive algorithms

Rule that “does work” then “calls itself” on a smaller version of the problem

Base case that handles the smallest problem

Prevents “infinite recursion”

Recursion example – print “hello” 5 times

Rule: Print “hello” once and then print “hello” 4 times

Base case: When the number of times to print is 0, stop printing

Recursive functions – base case

Conditional statement that prevents infinite repetitions

Usually handles cases where:

- input is empty

- problem is at its smallest size



Agenda

Recap

Recursion Examples

4 problems

Recursion Example - Factorial

$$n! = n * (n - 1) * (n - 2) * \dots * 1$$

$$3! = 3 * 2 * 1 = 6$$

$$4! = 4 * 3 * 2 * 1 = 24$$

Visualizing recursion – Factorial example

factorial(5) =

= 5 * factorial(4)

= 5 * 4 * factorial(3)

= 5 * 4 * 3 * factorial(2)

= 5 * 4 * 3 * 2 * factorial(1)

= 5 * 4 * 3 * 2 * 1

Recursion Example – Contains letter

Write a method called ContainsLetter.

Arguments: String (haystack), Character (needle)

Return: **true** is character is in the String, **false** otherwise

Recursion Visualization – Contains letter

```
contains("l", "apple") =  
    contains("l", "apple")  
        contains("l", "pple")  
            contains("l", "ple")  
                contains("l", "le", 3)  
                    return true
```

Recursion Example – IndexOf letter

Write a method called IndexOf.

Arguments: String (haystack), Character (needle)

Return: the index of the character in the String, if the character isn't there, return:

-1.

Recursion Example – printList

Write a recursive function that prints the contents of an array

Recursion limitations

- Limited number of times we can recurse
 - Stackoverflow – too many frames
- Potentially memory inefficient
 - If we copy data in subproblems – we'll worry about this in a few weeks
- Performance: might duplicate unnecessary work
 - We'll define performance later in the semester