

COSC 2306

Data Programming

Recursion

In-class practice

- Write a recursive function `pow2(n)` to compute 2^n (n is a non-negative integer)

Base case: $2^0 = 1$

General case: $2^n = 2 * 2^{n-1}$

2's Exponentials - recursive

```
def pow2_recursive(n):  
    if n==0: #base case  
        return 1  
    else:  
        return 2*pow2(n-1) #general case
```

In-class practice

- Write an iterative function `pow2(n)` to compute 2^n (n is a non-negative integer)

2's Exponentials - iterative

```
def pow2_iterative(n):  
    result = 1  
    for i in range(n):  
        result *= 2  
    return result
```

2's Exponentials - compare

```
def pow2_iterative(n):  
    result = 1  
    for i in range(n):  
        result *= 2  
    return result
```

```
def pow2_recursive(n):  
    if n==0: #base case  
        return 1  
    else:  
        return 2*pow2(n-1) #general case
```