

Mathematical Software Programming (02635)

Module 11 — Fall 2016

Instructor: Martin S. Andersen

Announcements

Course evaluation

Survey opens today (November 17) and closes on November 28 at midnight.

Your feedback is valued

- ▶ What activities/exercises helped you learn the material?
- ▶ Which concepts/exercises/lectures/assignments/... were difficult?
What can we do to improve?
- ▶ Is there anything that you expected to learn in this course but did not?
- ▶ Do you feel that your programming skills have improved throughout the course?

This week

Topic

- ▶ Recursion

Learning objectives

- ▶ Compare iterative and recursive solutions for simple problems
- ▶ Analyze the runtime behaviour and the time and space complexity of simple programs

Recursive functions

Definition

A recursive function is a function that calls itself during its execution

Example 1: Factorial (single recursion)

Base case: $f_0 = 1$

Recursive case:

$$f_n = n \cdot f_{n-1}, \quad n \geq 1$$

Example 2: Fibonacci numbers (multiple recursion)

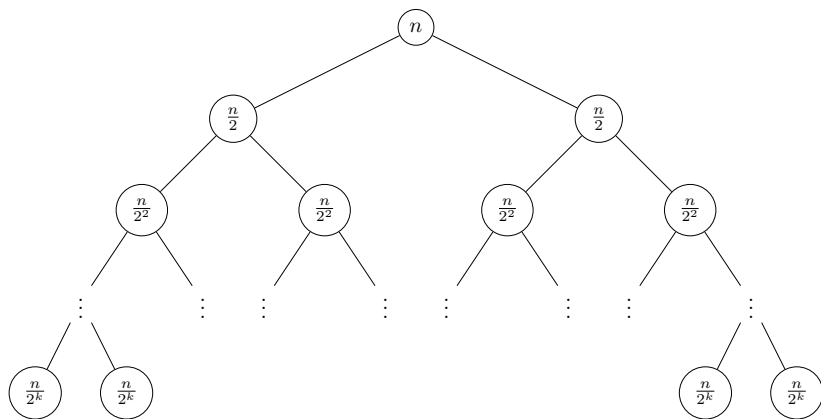
Base cases: $f_0 = 0$ and $f_1 = 1$

Recursive cases:

$$f_n = f_{n-1} + f_{n-2}, \quad n \geq 2$$

Divide and conquer

Break problem into subproblems and combine answers



Example: power function

The function x^n (with $n > 0$ and integer) can be expressed as

$$x^n = \begin{cases} x^{n/2} \cdot x^{n/2} & n \text{ even} \\ x \cdot x^{(n-1)/2} \cdot x^{(n-1)/2} & n \text{ odd} \end{cases}$$

Example: power function

Non-recursive implementation of power function x^n ($n \geq 0$ integer)

```
double power_v1(double x, unsigned int n) {  
    double val = 1.0;  
    for (int i=0; i<n; i++) val *= x;  
    return val;  
}
```

What is the space/time complexity?

Example: power function

Recursive implementation of power function x^n ($n \geq 0$ integer)

```
double power_v2(double x, unsigned int n) {  
    double val;  
    if (n == 0)  
        return 1.0;  
    val = power_v2(x, n/2);  
    if (n%2 == 0)    // n is even  
        return val*val;  
    else            // n is odd  
        return x*val*val;  
}
```

What is the space/time complexity?

Example: power function

Non-recursive implementation of power function x^n ($n \geq 0$ integer)

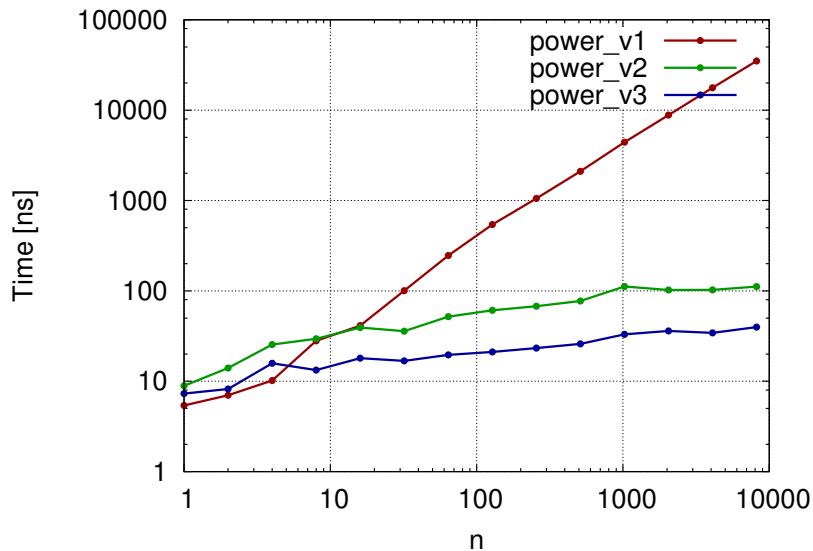
```
double power_v3(double x, unsigned int n) {  
    double val = 1.0;  
    while(n != 0){  
        if(n%2 == 0) {    // n is even  
            x = x*x;  
            n = n/2;  
        }  
        else {            // n is odd  
            val = val*x;  
            n = n-1;  
        }  
    }  
    return val;  
}
```

What is the space/time complexity?

Complexity

Function	Space complexity	Time complexity
power_v1	$O(1)$	$O(n)$
power_v2	$O(\log n)$	$O(\log n)$
power_v3	$O(1)$	$O(\log n)$

Experiment



Quiz 3

1. Go to socrative.com on your laptop or mobile device
2. Enter “room number” **02635**
3. Answer ten quick question (the quiz is anonymous)