Introduction

Wei-Mei Chen

Department of Electronic Engineering
National Taiwan University of Science and Technology

Data Structures

Data Structures

- Niklus Wirth

  Algorithms + Data structures = Programs

- Textbook:
  Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, *Fundamentals of Data Structures in C*
- Grade:
  - Assignments: 40%
  - Midterm Exam: 20%
  - Final Exam: 20%
  - Participation: 20%

Schedules

1. Basic Concepts
2. Arrays and Structures
3. Stacks and Queues
4. Lists
5. Trees
6. Graphs
7. Sorting

Assignments

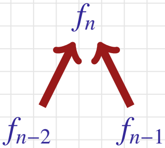
1. Cover
2. Contents (numbers of pages)
3. Problem, Definition, and Details (methodology and explanation)
4. Code
5. Results
6. Discussion and Conclusion

Fibonacci Sequence

$$f_n = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ f_{n-2} + f_{n-1} & \text{else} \end{cases}$$

defined recursively!

$0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$



f_n

f_{n-2} f_{n-1}

divide-and-conquer approach

Fibonacci Sequence I

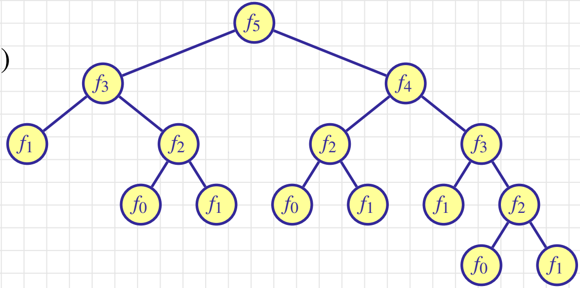
```
int fib1(int n) {
    if (n<=1)
        return n;
    else return fib1(n-1)+fib1(n-2);

}
```

n	# of Terms
0	1
1	1
2	3
3	5
4	9
5	15
6	25

If n is a power of 2

$$\begin{aligned} T(n) &> T(n-1) + T(n-2) \\ &> 2 \cdot T(n-2) \\ &> 2 \cdot 2 \cdot T(n-4) \\ &> \cdots \\ &> 2 \cdot 2 \cdots 2T(0) \end{aligned}$$

$$\Longrightarrow T(n) > 2^{n/2}$$



Fibonacci Sequence II

```
int fib2(int n) {
    int i;
    int f[n+1]=0;
    if (n>0) {
        f[1]=1;
        for(i=2; i<=n; i++)
            f[i]=f[i-1]+f[i-2];
        };
    };
    return f[n];
}
```

Recompute the same value $\Longrightarrow$ Save it in an array

Linear Time: $O(n)$

fib1 vs. fib2

Assume that one term can be computed in 10^{-9} sec.
1 ns = 10^{-9} sec and 1μs = 10^{-6} sec

n	$2^{n/2}$	Execution Time by fib2	Execution Time by fib1
40	1048576	41 ns	1048 μs
60	1.1×10^9	61 ns	1 s
80	1.1×10^{12}	81 ns	18 min
100	1.1×10^{15}	101 ns	13 days
120	1.2×10^{18}	121 ns	36 years
160	1.2×10^{24}	161 ns	3.8×10^7 years
200	1.3×10^{30}	201 ns	4×10^{13} years