

绪论

计算：算法

01-A2

Computer science should be called computing science, for the same reason why surgery is not called knife science.

- E. Dijkstra

邓俊辉

deng@tsinghua.edu.cn

# 算法

❖ 计算 = 信息处理 = 借助某种工具，遵照一定规则，以明确而机械的形式进行

❖ 计算模型 = 计算机 = 信息处理工具

❖ 所谓算法，即特定计算模型下，旨在解决特定问题的指令序列

输入 待处理的信息（问题）

输出 经处理的信息（答案）

正确性 的确可以解决指定的问题

确定性 任一算法都可以描述为一个由基本操作组成的序列

可行性 每一基本操作都可实现，且在常数时间内完成

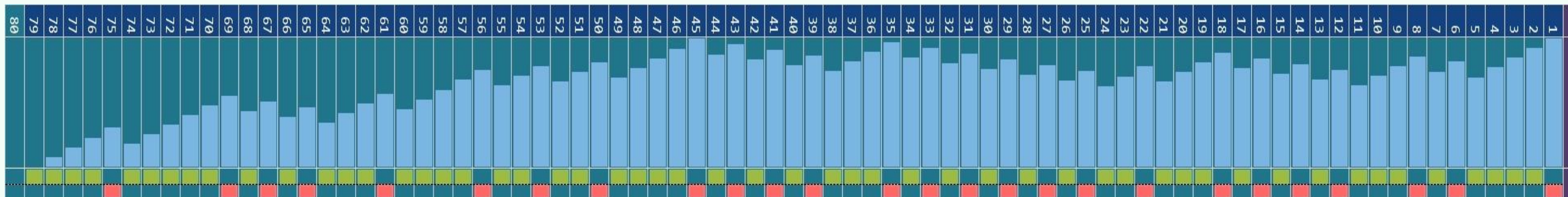
**有穷性** 对于任何输入，经有穷次基本操作，都可以得到输出

# 有穷性：Hailstone序列

$$Hailstone(n) = \begin{cases} \{1\} & (n \leq 1) \\ \{n\} \cup Hailstone(n/2) & (n \text{ is even}) \\ \{n\} \cup Hailstone(3n + 1) & (n \text{ is odd}) \end{cases}$$

42, 21, 64, 32, ..., 1

7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, ..., 1



27, 82, 41, 124, 62, 31, 94, 47, 142, 71, 214, 107, ...

# 有穷性：程序~算法

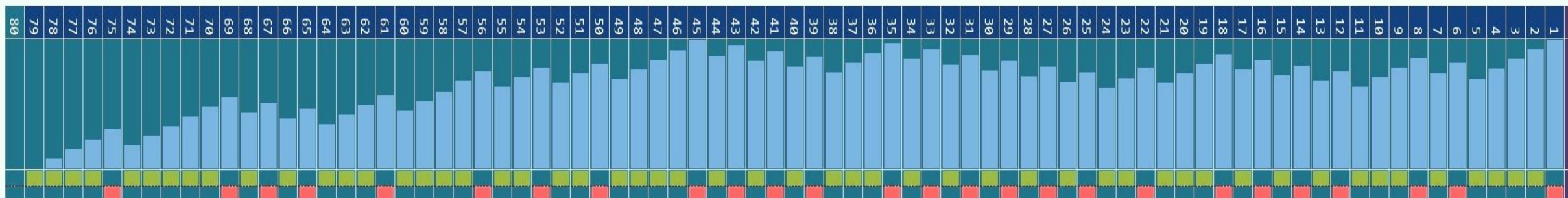
```
❖ int hailstone( int n ) {
```

```
    int length = 1;
```

```
    while ( 1 < n ) { n % 2 ? n = 3*n + 1 : n /= 2; length++; }
```

```
    return length;
```

```
} //对于任意的n , 总有|Hailstone(n)| < ∞ ?
```



❖ Erdos: Mathematics is **NOT** yet ready for such problems



# 好算法 = ?

❖ 正确：符合语法，能够编译、链接

能够正确处理简单的输入

能够正确处理大规模的输入

能够正确处理一般性的输入

能够正确处理退化的输入

能够正确处理任意合法的输入

❖ 健壮：能辨别不合法的输入并做适当处理

而不致非正常退出

❖ 可读：结构化 + 准确命名 + 注释 + ...

❖ 效率：速度尽可能快

存储空间尽可能少

**Algorithms + Data Structures = Programs**

**(Algorithms + Data Structures) x Efficiency = Computation**