

Algorithms

Introduction

(Based on [Manber 1989])

Yih-Kuen Tsay

Dept. of Information Management
National Taiwan University



What They Are

- 🌐 An **algorithm** is, broadly speaking, a *step-by-step* procedure for solving a problem or accomplishing some end.
- 🌐 When it is meant for the **computer**, each step in an algorithm should be realizable by *well-defined*, limited *primitive* operations that the computer understands.
- 🌐 **Algorithm design** is an important and usually the hardest part of **programming** (which consists in finding/devising a solution and translating it into a computer program).
- 🌐 Better algorithms (designed once, used forever) save more time and money.



Development of an Algorithm

- 🌐 We typically are given a problem statement, including **input** and **output** requirements, that is an abstract yet *accurate* and *precise* account of the problem to be solved and the properties of a satisfactory solution.
- 🌐 The development of an algorithm involves the following tasks:
 - ☀️ **Design** (main subject of this course)
 - ☀️ **Verification** (or **Proof of Correctness**)
 - ☀️ **Analysis**
 - ☀️ **Implementation**



Main Concerns

- 🌐 Why is algorithm design difficult?
 - ☀️ Counterintuitive approaches may be needed, because of large problem scales.
 - ☀️ Better solutions, if worthwhile, may be more complicated.
- 🌐 How do we approach it?



Our Approach to the Subject

- 🌍 Two distinct features:
 - ☀️ Emphasis of the **creative** side
 - 👤 learning to create by trying to create
 - ☀️ **Induction** as one central design method
 - 👤 to explain/understand the principles behind a design
 - 👤 to systematically guide the creation process
- 🌍 What is the “design by induction” method?
 - ☀️ draw analogies from proving theorems by mathematical induction
 - ☀️ concentrate on **extending** solutions of smaller problems instances to solutions of larger ones
 - ☀️ induction may not solve every problem, but is helpful

