CS140

• Who Am I?
• Who Are You?
• Course Overview
• Some Basics

Course Overview

1. How to analyze algorithms
2. How to design algorithms
3. NP-completeness

Course Requirements

• Homework
• Exams
• Class participation

Course Requirements - Homework

• Posted on Monday and Wednesday
  (Yes! Two assignments each week.)
• Monday assignment due Wednesday at noon,
  Wednesday assignment due Friday at noon
• Solutions will be posted at due time so no late
  homework will be accepted
• Your 3 lowest homework grades will be dropped
• Solutions should be prepared in LaTeX

Course Requirements – Exams

(Tentative Schedule)

• Exam I: Wed. 9/25
• Exam II: Wed. 9/8
• Final:

• Exams will be in-class, closed book

Course Requirements – class participation

• Show up to class
• Speak up in class
• Hand in daily “worksheets”
Advice

• Stay on top of things
• Seek help when necessary

The Problem

Computational Problem: Specified by input/output pair

Sorting

• Sorting Integers in Ascending Order (SIAO):
  Input: A list of integers
  Output: The input integers sorted in ascending order

• Example:
  Input: 5,3,8,1,2
  Output:

The Algorithm

• Computational Problem: Specified by input/output pair.

• Algorithm: Well-defined sequence of computational steps that produce a correct output for every valid input.

What should an algorithm for SIAO do?

• Example 1:
  – Input: 5,3,8,1,2
  – Output: 1,2,3,5,8

• Example 2:
  – Input: 3,a,5.27,mudder
  – Output: we don’t care

An Algorithm for SIAO?

Sort1(S)

While there are integers x and y in S such that x precedes y in S and x > y
Swap x and y in S
Return S
Sort1 Example

Input: 5, 3, 8, 1, 2
Swap 3 and 2: 5, 2, 8, 1, 3
Swap 3 and 5: 3, 2, 8, 1, 5
Swap 3 and 2: 3, 2, 8, 1, 5
Swap 2 and 1: 1, 3, 8, 2, 5
Swap 3 and 2: 1, 2, 8, 3, 5
Swap 8 and 5: 1, 2, 8, 3, 5
Swap 5 and 3: 1, 2, 3, 5, 8
Stop

Sort1 Example cont.

Input: 5, 3, 8, 1, 2
Swap 3 and 2: 5, 2, 8, 1, 3
Swap 3 and 5: 3, 2, 8, 1, 5
Swap 3 and 2: 2, 3, 8, 1, 5
Swap 2 and 1: 1, 3, 8, 2, 5
Swap 3 and 2: 1, 2, 8, 3, 5
Swap 8 and 5: 1, 2, 5, 3, 8
Swap 5 and 3: 1, 2, 3, 5, 8
Stop

Is Sort1 an algorithm?

Is Sort1(S) well-defined?

No! We need to specify a selection rule.

Software Development

The problem: Huh?
The idea: A-ha!
The program: Ta-da!

The algorithm exists somewhere between a-ha and ta-da.

An Algorithm for SIAO?

Sort2(S)
Assume a fixed order on pairs of elements in S
While there are integers x and y in S such that x precedes y in S and x > y
Choose first pair x, y that is out of order
Swap x and y in S
Return S

Is Sort2 an algorithm?

Is Sort2(S) well-defined? Yes.

Does it produce the correct output for any valid input?
Proof of correctness

- When the algorithm halts, S is sorted.
- The algorithm halts on all input.

Claim: The number of out-of-order pairs in S decreases with each swap.

Sort2 Example revisited

Input: 5, 3, 8, 1, 2

Swap 3 and 2: 5, 2, 8, 1, 3
Swap 3 and 5: 3, 2, 8, 1, 5
Swap 3 and 2: 2, 3, 8, 1, 5
Swap 2 and 1: 1, 3, 8, 2, 5
Swap 3 and 2: 1, 2, 8, 3, 5
Swap 8 and 5: 1, 2, 5, 3, 8
Swap 8 and 3: 1, 2, 3, 5, 8
Stop

Is Sort2 a good algorithm?

- Is it easy to understand?
- Is it easy to implement?
- Is it fast?
- Is it space-efficient?

How fast is Sort2?

- At most $n^2$ comparisons are used to find an out of order pair.
- At most $n^2$ pairs are initially out of order.

Sort2 is $O(n^4)$.

CS140: Two questions

Is it correct?  Is it fast?

Computational procedure $\rightarrow$ Algorithm $\rightarrow$ Good algorithm