

CH-231-A

Algorithms and Data Structures

ADS

Lecture 6

Dr. Kinga Lipskoch

Spring 2024


Definition: Algorithm

- ▶ An algorithm is a sequence of computational steps which transforms a set of values (input) to another set of values (desired output)
- ▶ It is a tool for solving a well-defined computational problem
- ▶ Step-wise procedure that can be implemented in a computer program
- ▶ Consists of a finite list of well-defined instructions (Turing machine)
- ▶ 'Algorithm' stems from 'Algoritmi', the Latin form of al-Khwārizmī, a Persian mathematician, astronomer and geographer

Example: Sorting Problem

- ▶ Input
sequence $\langle a_1, a_2, \dots, a_n \rangle$ of numbers
- ▶ Output
permutation $\langle a'_1, a'_2, \dots, a'_n \rangle$
such that $a'_1 \leq a'_2 \leq \dots \leq a'_n$
- ▶ **Example** (instance of sorting problem):
Input: 8 2 4 9 3 6
Output: 2 3 4 6 8 9


Example: Searching



[All](#)
[Videos](#)
[Images](#)
[Books](#)
[Shopping](#)
[More](#)
[Settings](#)
[Tools](#)

About 34,100,000 results (0.50 seconds)

Learn about arrays, linked lists, binary trees, **hash tables**, graphs, stacks, queues, heaps, and other fundamental data structures. Sep 13, 2017



Improving your Algorithms & Data Structure Skills – Coderbyte – Medium
<https://medium.com/.../how-to-get-good-at-algorithms-data-structures-d33d5163353f>

About this result Feedback

Data Structures and Algorithms | Coursera
<https://www.coursera.org/specializations/data-structures-algorithms>

This specialization is a mix of theory and practice: you will learn algorithmic techniques for solving various computational problems and will implement about 100 algorithmic coding problems in a programming language of your choice. No other online course in Algorithms even comes ...

Algorithms and Data Structures - edX
<https://www.edx.org/course/algorithms-data-structures-microsoft-dev285x-1>

Want to build better programs? Learn how, in this professional-level course. Bring your programming experience, and join us for a deep dive into fundamental concepts that you can use right away. Go underneath the hood of functional algorithms and data structures, and see how they work and how to compare them. Plus ...

Algorithms and Data Structures | edX
<https://www.edx.org/micromasters/ucsan-diego-x-algorithms-and-data-structures>

A series of credit-eligible courses recognized by industry. This MicroMasters program is a mix of theory and practice: you will learn algorithmic techniques for solving various computational problems through implementing over one hundred algorithmic coding problems in a programming language of your choice.

Improving your Algorithms & Data Structure Skills – Coderbyte – Medium
<https://medium.com/.../how-to-get-good-at-algorithms-data-structures-d33d5163353f>

Sep 13, 2017 - Learn about arrays, linked lists, binary trees, hash tables, graphs, stacks, queues,

Algorithms + Data Structures = Programs


Book by Niklaus Wirth

89% liked this book
Google users


Algorithms + Data Structures = Programs is a 1978 book written by Niklaus Wirth covering some of the fundamental topics of computer programming, particularly that algorithms and data structures are inherently related. [Wikipedia](#)

Originally published: 1976
Author: Niklaus Wirth


People also search for




Introduction to Algorithms
Charles E. Leiserson




Data Structures and Algorithms
Robert Tamassia



The Art of Computer Programming
Donald Knuth



Data structures and algorithms
Michael T. Goodrich

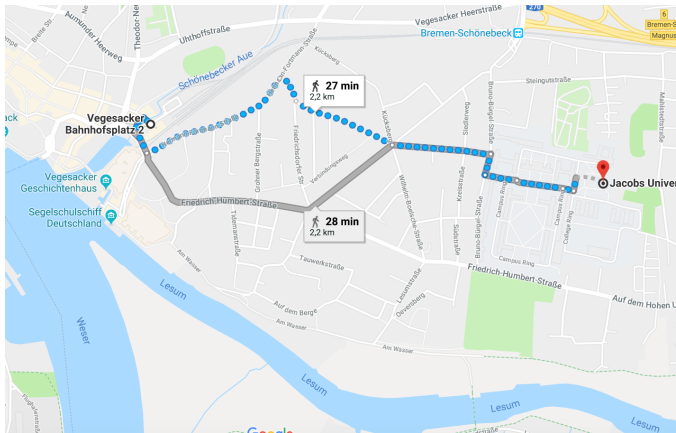


Fundamentals of data structures
Elias Horowitz

View 60+ more

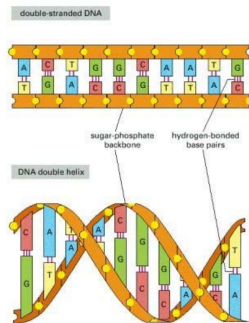
Feedback

Example: Road map



Graph algorithm

Example: DNA Sequences



String matching

Analysis of Algorithms

- ▶ The theoretical study of computer-program performance and resource usage
- ▶ Other design goals?
 - ▶ correctness
 - ▶ functionality
 - ▶ robustness
 - ▶ reliability
 - ▶ user-friendliness
 - ▶ programmer time
 - ▶ simplicity
 - ▶ modularity
 - ▶ maintainability
 - ▶ extensibility

Performance of Algorithms

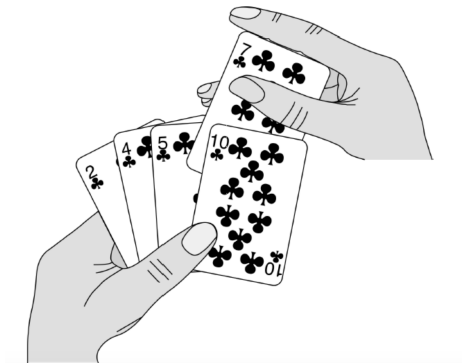
- ▶ Analysis helps us to understand scalability
- ▶ Performance often draws the line between what is feasible and what is impossible
- ▶ Algorithmic mathematics provides a language for talking about program behavior
- ▶ "Performance is the currency of computing"
- ▶ The lessons of program performance generalize to other computing resources

Definition: Data Structure

- ▶ A data structure is a way to store and organize data in order to facilitate access and modification
- ▶ There is typically no best data structure, but each data structure has its strengths and weaknesses
- ▶ Which data structure to use, depends on the problem that is to be solved
- ▶ Sometimes there is a trade-off between storage (in a data structure) and speed (in accessing a data structure or of an algorithm)

Sorting Problem

First algorithm: Insertion sort



Insertion Sort

INSERTION-SORT(A, n)

for $j = 2$ **to** n

$key = A[j]$

 // Insert $A[j]$ into the sorted sequence $A[1 \dots j - 1]$.

$i = j - 1$

while $i > 0$ and $A[i] > key$

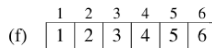
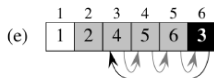
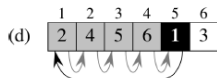
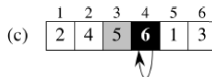
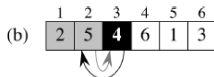
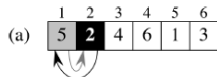
$A[i + 1] = A[i]$

$i = i - 1$

$A[i + 1] = key$

Insertion Sort: Example

Sort $A = \langle 5, 2, 4, 6, 1, 3 \rangle$



INSERTION-SORT(A, n)

for $j = 2$ **to** n

$key = A[j]$

 // Insert $A[j]$ into the sorted sequence $A[1 \dots j - 1]$.

$i = j - 1$

while $i > 0$ and $A[i] > key$

$A[i + 1] = A[i]$

$i = i - 1$

$A[i + 1] = key$