# Data Structures & Algorithms

Adil M. Khan
Innopolis University

a.khan@innopolis.ru

# About Me:

- Adil Khan, Ph.D.
- Full Professor
- Institute of Data Science and Artificial Intelligence
- Office: 466
- Email: a.khan@innopolis.ru

- Research interests:
  - ➢ Machine Learning, Deep Learning, Representation Learning

# Objectives

- Learn what are algorithms, data, and data structures
- Understand why we should learn about them?
- Know the course outline and related info

# Algorithms

# Algorithms

An Algorithm is a finite set of instructions that, if followed, accomplishes a general, well-specified task

- Input : A value or a set of values that are externally supplied.
- Output : A value or a set of value is produced.
- Definiteness : Each instruction is clear and unambiguous.
- Finiteness : The algorithm terminates after a finite number of steps/instructions.

# Algorithms

Three desirable properties of a good algorithms are

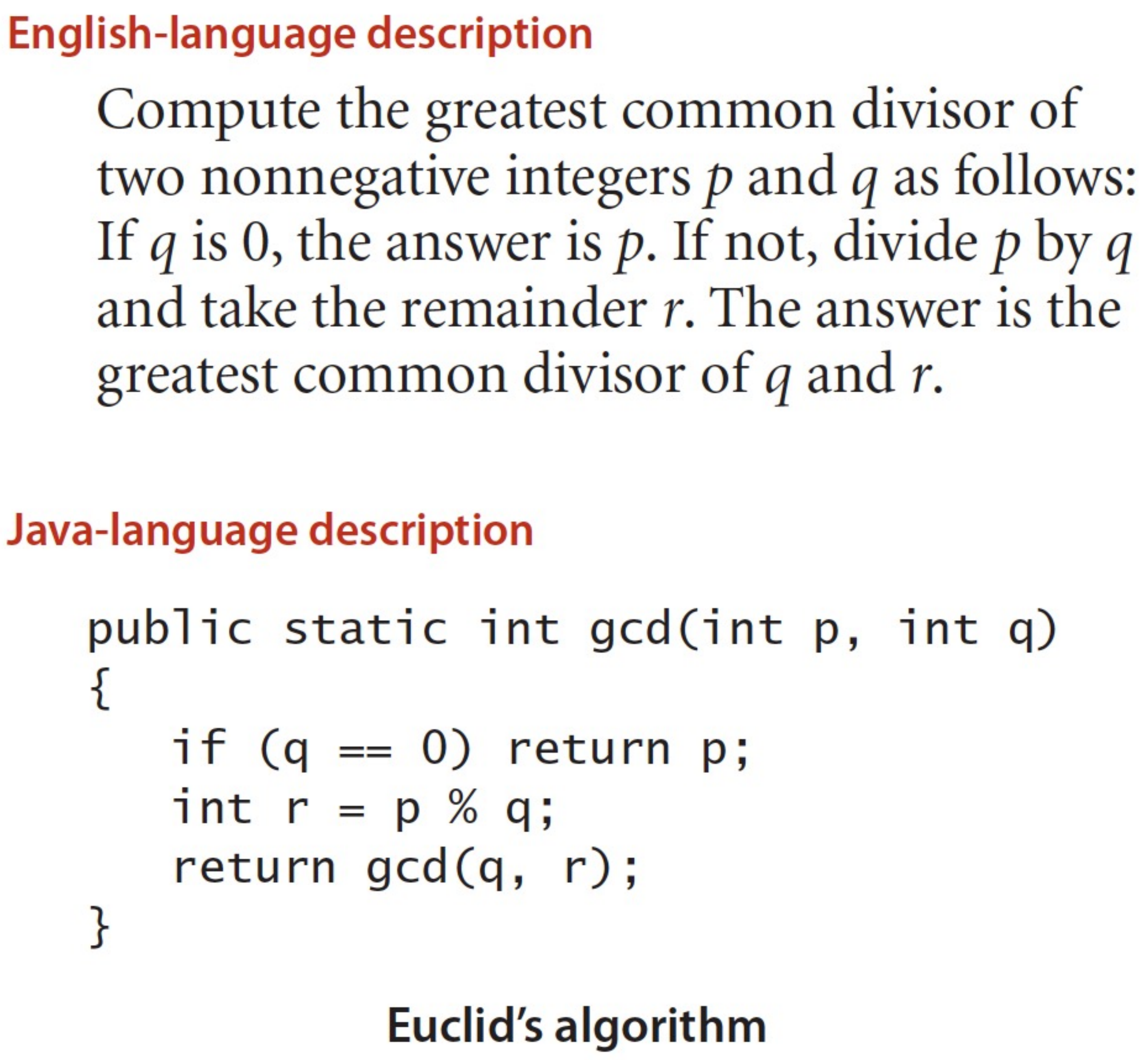
① Correctness

② Efficiency

③ Ease of implementation

# Algorithms

Can be described in a natural language or by writing a computer program



**English-language description**

Compute the greatest common divisor of two nonnegative integers *p* and *q* as follows: If *q* is 0, the answer is *p*. If not, divide *p* by *q* and take the remainder *r*. The answer is the greatest common divisor of *q* and *r*.

**Java-language description**

```
public static int gcd(int p, int q)
{
   if (q == 0) return p;
   int r = p % q;
   return gcd(q, r);
}
```

**Euclid's algorithm**

# Classifying Algorithms

- By ***Problem Domain:*** numeric, text processing, sorting, searching, networks, machine learning, …
- By ***Design Strategy:*** divide and conquer, greedy, dynamic programming, backtracking, …
- By ***Complexity:*** constant, linear, quadratic, cubic, exponential, …
- By ***Implementation Dimensions:*** sequential, parallel, recursive, iterative, …

# Data

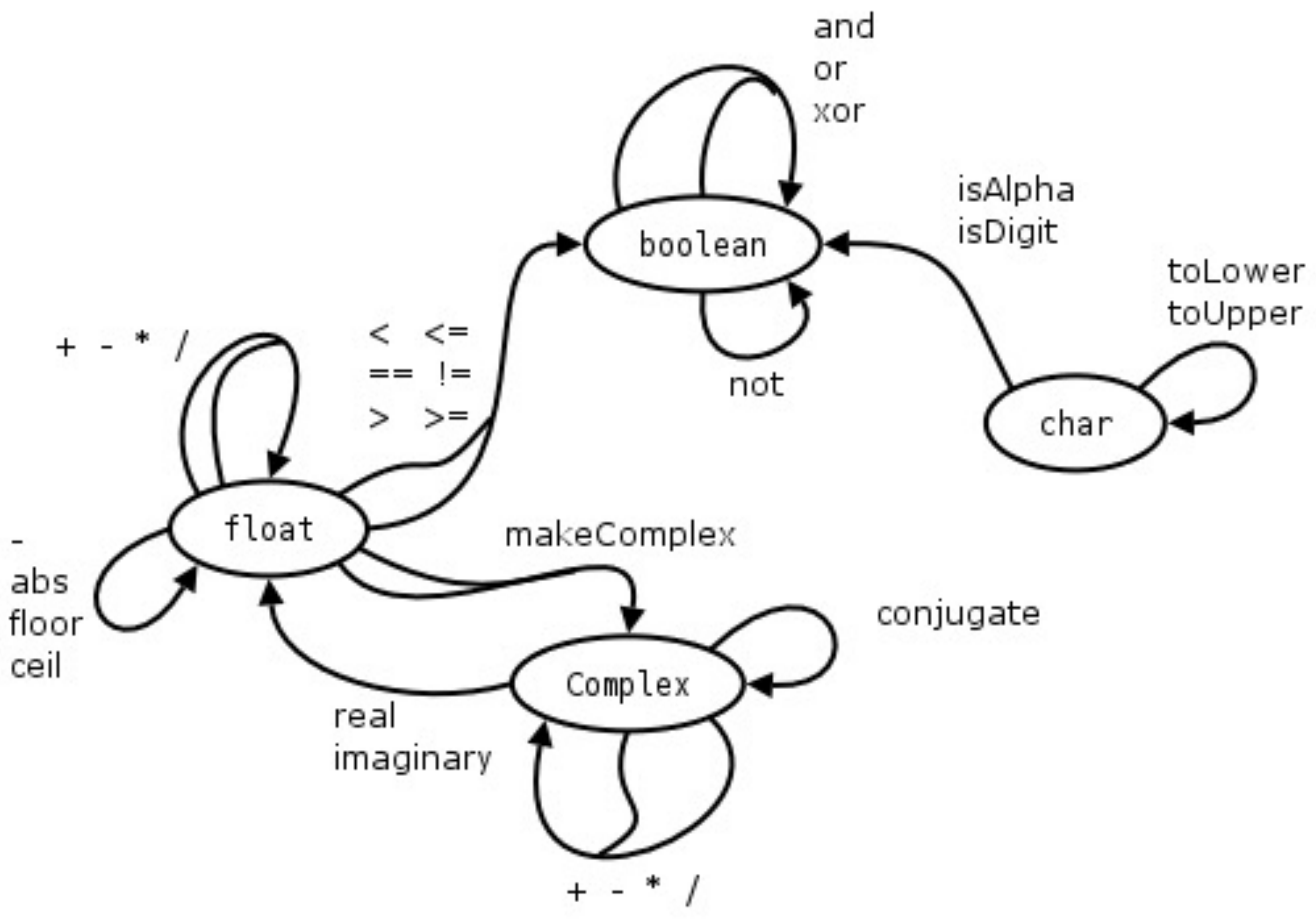
# Data

- Set of values

| | |
|---|---|
| Z | {. . ., -2, -1, 0, 1, 2, . . .} |
| N | {0, 1, 2, 3, 4, . . .} |
| B | {false, true} |

# Data Type

Data Type = Type + Operations



- Data Types
  - Primitive (value)
    - integer, float, boolean, character
    - string*, pointer*
  - Complex (reference)
    - employee, department, list, stack,… .

# Data Structures

- Arrangement of data for the purpose of being able to store and retrieve information

- Example: List, BST, Hash Table, etc.

# So, Why Study All of These?

- Because we want to produce efficient software, one which minimizes these
  - Time
  - Space (memory)
  - Coding Time
  - Verification and Debugging Time
  - System Integration Time

# Thus!

Computer Programs = Algorithms + Data Structures

*“Thus to produce an efficient program, we must design the right algorithm and choose the right data structure”*

# Algorithms

## (How to Choose The Right One)

- An art: requires cleverness, ingenuity, and sometimes dumb luck

- A Science: Principles of algorithm analysis, and widely applicable algorithm patterns have been developed over time

# Data Structures

## (Importance of Choosing The Right One)

- Changing the DS in a slow program can work the same way as an organ transplant does in a sick patient
- Has nothing to do with the correctness of the program
- Remember, it is better to be born with a good heart than have to wait for replacement
- For max benefit, choose the right data structure and design your program around it

# Data Structures

## (How to Choose The Right One)

- Some important questions to ask:
  - Can the DS be completely filled at the beginning, or will there be insertions along with deletions, lookups, updates and other operations?
  - Will the items be processed in a well-defined order, or will random access have to be supported?

# And that is what you will learn in this course, hopefully!

Computer Programs = Algorithms + Data Structures

*“Thus to produce an efficient program, we must design the right algorithm and choose the right data structure”*

# Course Structure

# Grading

- Scale Rounded
  - A: 90% +
  - B: 75% +
  - C: 60% +
  - D: Everyone else

# Other Info

- Bonus Points (Each TA has 5% Bonus GRADE points for each group)
- Attendance Policy
- Cheating Policy