Data Structures
Computers 303B

Prof. Dr. Khaled Fouad Elsayed

Course Information

- Instructors:
  - Prof. Dr. Khaled Fouad Elsayed
  - Dr. Omar Nasr
  - Teaching Assistant: Eng. Yossif
- Office Hours for Dr. KFE: Sunday 12-2 PM
- Textbooks:
  - Data Structures and Algorithms in C++: Adam Drozdek
  - Data Structures via C++: M. Berman

Course Contents/Schedule

3

Topic	Lectures
Introduction + Software Engineering + Abstraction	1
C++ Classes (ADT)	3
The List ADT	4-5
Queues and Stacks	3
Searching and Sorting	3
Tables and Fast Searching	2
Trees	2

- Midterm after Lists or at the end

Grading

4

- 70% Final Exam
- Midterm: 13-16 Points
- 4 Programming Assignments: 17-14 Points
- First Assignment distributed this week

Some good resources

5

- C++
  - http://www.cplusplus.com/doc/tutorial
  - https://softwareengineering.stackexchange.com/
  - https://stackoverflow.com/questions/tagged/c%2b%2b
- More advanced
  - https://www.codeproject.com/
  - http://en.cppreference.com

So why teach E&C Engineers Data Structures?

6

- Ask department graduates ☺
  - Opens best jobs in the market in and outside Egypt for those who tried their best
- Data Structures is basis for serious programming
- Learn an indispensable skill for an Elec/Comm Eng
  - Serious Embedded Systems (e.g. communication protocols implementation)
  - EDA software
  - Machine learning/Data Analytics/….
  - Hardware design IS now software-based
  - …….
- Innovate

Cairo University

7

- Consider the course as a workshop

Get your hands dirty → The more the better

8

Increasing importance of software

What are the world's most valuable companies?

9

Ranking of the companies rank 1 to 100	Market value in billion U.S. dollars
Apple	926.9
Amazon.com	777.8
Alphabet	766.4
Microsoft	750.6
Facebook	541.5
Alibaba	499.4
Berkshire Hathaway	491.9
Tencent Holdings	491.3
JPMorgan Chase	387.7
ExxonMobil	344.1
Johnson & Johnson	341.3
Samsung Electronics	325.9
Bank of America	313.5

May 2018

What are the world's most valuable companies?

10

Apple	1	692.8
Alphabet (Google)	2	573.8
Microsoft	3	489.5
Berkshire Hathaway	4	402.0
Amazon.com	5	391.0
Facebook	6	387.8
Exxon Mobil	7	338.0
Johnson & Johnson	8	308.5
JPMorgan Chase	9	306.9
Wells Fargo	10	279.0
General Electric	11	258.8
Alibaba Group	12	257.4
AT&T	13	253.0

Hot Topics

11

- Big Data/Analytics
- AI/Machine Learning
- Autonomous Driving
- Internet of Things (IoT)
- Business processes
- Mobile Apps
- Games

Software is pervasive throughout our economy and culture

12

- Modern civilization runs on software.
- Nearly all of the products, services and innovations that power the industrialized world depend on software.
- Software is largely invisible. You can't see it. You can only experience it indirectly.
- The Platform companies …

For Example

13

- Did you know there is a ton of software in the average luxury car?
- The average luxury car contains almost 100 million lines of code.
- Printed and bound, 100 million lines of code weighs over 1.5 tons!

- General Motors:
  - Typical high-end car can have as many as 65 microprocessors
  - Cost of a car's electronics (hardware and software combined) now exceeds that of material components such as steel, aluminum, and glass.

The Demand for Software is Strong and Growing

14

- Moore's Law
- Software replacing hardware
- Product differentiation with software
- Open Source
- Growing interest in mobile devices (smart phones, tablet computers, etc.) (Post-PC era)
- Autonomous Vehicles

15

Software Engineering

16

What is Engineering?

- Engineers apply science and technology to develop cost-effective solutions to practical problems.
- Engineering disciplines have a core body of knowledge or underlying science that can be used to solve practical problems.
- For example, chemical engineering has chemistry and electrical engineering has math and physics.
- Software Engineering?

Software Engineering

17

- Software engineering is “the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, that is, the application of engineering to software” [IEEE]

Software Engineering vs Computer Science

18

- Computer science focuses on theory and fundamentals
- Software engineering is concerned with the practicalities of developing and delivering useful software

19

Program Design

Importance of Design

20

- Complex programs need to be designed and analysed before coding starts.
- Don’t Start by Coding
- The Seduction of screen and keyboard
- Software is hard ---Donald Knuth
- Typical design process is longer than development
  - Specification: WHAT system does NOT HOW it is done
  - Detailed design: Describe how a certain system achieves specs
- Most projects experience a 4-2-4 time plan
  - 4 Design, 2 Coding, 4 Testing (or even 3-1-6)

Software complexity can run out-of-control

21

- Usually used to automate some process or solve a problem
- → Complexity is “built-in”
- Adding lines is easy and can be without boundaries (not the case in other engineering systems)
- Complexity increases as team size increases

Means to manage complexity

22

- Abstractions (Modeling, more later)
- Decomposition: Divide and Conquer

23 Abstractions as a Means to Manage Complexity

Data Structures == Abstract Data Types

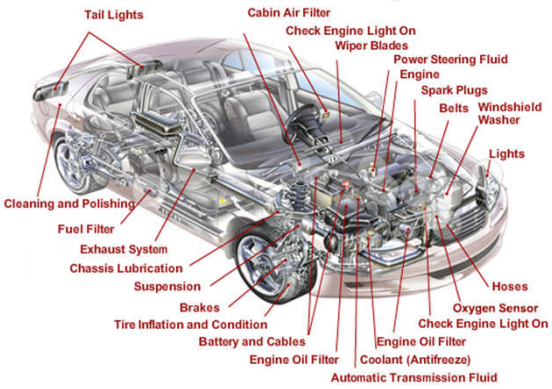
24

- So, what is abstractions all about?
- Abstraction: A model of a complex system that captures only the necessary details.
- Different abstractions are used by different stakeholders of the system.

Complex Systems Example: Modern Car

25

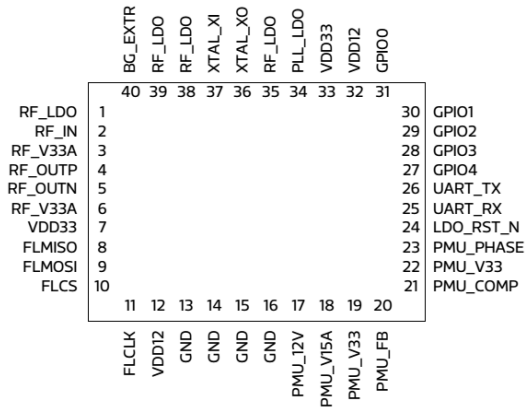
- Driver cannot know all details of complex working of the car.



IC

26

- You use the IC by reading datasheet and knowing what each PIN does.
- Then you build a larger system out of this
- This is abstraction
- Another engineering team designs the IC



Abstractions in SW Design

27

- In SW design, modules hide internal complexity and provide external interface through which its functionality can be used.
- Information hiding/data abstraction: hiding details of a function or data structure with the goal of controlling access to the details of a module/structure.
- Software design process must help us divide a complex system into smaller, more manageable modules easier to build.
- Modules must be:
  - Good abstraction
  - Cohesive

28

- Each module should have a single purpose of identity.
- Modules should stick together well.
- Changes in a module don't result in changes in modules that use it.
- Good exercise:
  - Identify all modules used in a mobile phone system

Homework

29

- Review C++ from 1st year
- Watch these videos
  - goo.gl/GXc5FP
  - goo.gl/lpqX6k