CS309
OBJECT-ORIENTED DESIGN

Yuqun Zhang （张煜群）
Department of Computer Science and Engineering
Southern University of Science and Technology

Who and Where Am I?

- Dr. Yuqun Zhang （张煜群）
- Email: zhangyq@sustc.edu.cn
- Office: Room 1007, 智园A7 (Soon it is going to be Room 604, 创园4栋)
- Office Hours: 4-6pm, Thursday, or appointment by email

A LITTLE SOMETHING ABOUT ME…

My Styles and Rules

- Interaction
- Mutual Respect
- NO CHEATING!!!!!
  - You may work together in this class, as specified on each specific assignment. Do **NOT** use any resources without citation.

Teaching Assistant

- Yueming Zhu
  - Email: zhuym@sustc.edu.cn
- Mingyuan Wu
- Yang Chen
- Weiduo Liao

Student Assistant

- Zhihao Dai
- Chao Pan
- Zhengchang Hua

Textbooks

- Freeman et al., *Head First Design Patterns*
- Block, *Effective Java*
- Zeller and Krinke, *Essential Open Source Toolset: Programming with Eclipse, JUnit, CVS, Bugzilla, Ant, Tcl/TX and More*
- McConnell, *Code Complete: A Practical Handbook of Software Construction*
- Barrett, *Linux Pocket Guide*
- Pilone, *UML 2.0 Pocket Reference*

Evaluation and Grading

- Weekly Lab Tutorials– 25%
  - ~12 times
- Project – 35%
  - Web applications
  - Group of 5 (before the 1st lab next week)
  - 3 presentations (proposal, progress, final)
  - 2 written reports
- Exams – 30%
  - Final:
    - What's on an exam? Anything from any aspect of class, including lab sections.
    - No hints (重点）
- In-Class Exercises/Attendance – 10%
  - Spontaneous

ALRIGHT, LET'S GET REAL

Expectations

- You're going to have to "own" your education in this class
  - I have a feeling this is going to be an awesome semester…
- But…
  - Expect that I may not be able to give you an immediate answer (I'm alright if my response to your question is "I don't know," so you're going to have to be alright with that, too)
  - I (or the TAs) WILL always try to help find you the answers you need in a timely fashion. Be patient.

Tips of Handling Problems

- Once you encounter problems (theory or practice), you are expected to
  - first, try your real best to solve them by yourself

Tips of Handling Problems

- Once you encounter problems (theory or practice), you are expected to
  - first, try your real best to solve them by yourself
  - if not working, try to talk with your cohorts.

Tips of Handling Problems

- Once you encounter problems (theory or practice), you are expected to
  - first, try your real best to solve them by yourself
  - if not working, try to talk with your cohorts.
  - if not working, then ask us

Tips of Handling Problems

- Once you encounter problems (theory or practice), you are expected to
  - first, try your real best to solve them by yourself
  - if not working, try to talk with your cohorts.
  - if not working, then ask us
- If we find that you are not paying effort by yourself, we would be reluctant to help you at later time.


The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, you may have to delete the image and then insert it again.

What You Would Learn

- Of course the object-oriented design and analysis
- Typically, you are going to learn something about
  - requirement engineering (UML)

UML Examples

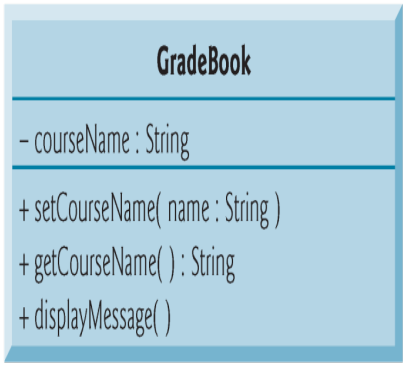


Fig. 7.3 | UML class diagram for class `GradeBook`.

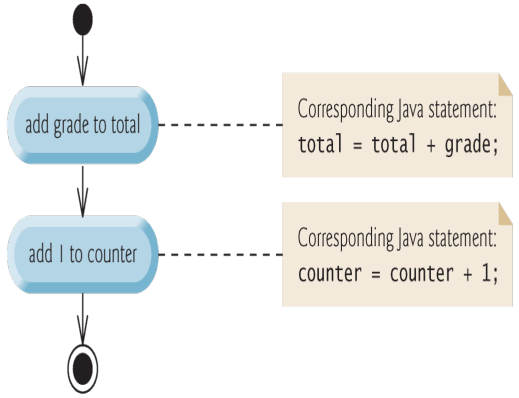


Fig. 3.1 | Sequence structure activity diagram.

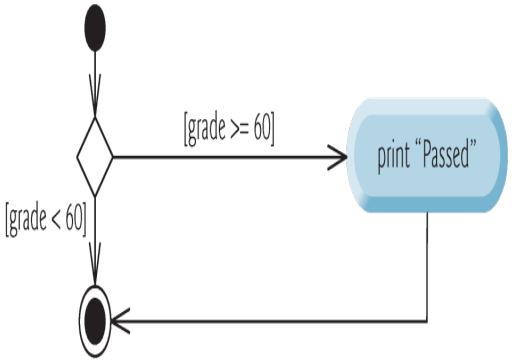


Fig. 3.2 | `if` single-selection statement UML activity diagram.

What You Would Learn

- Of course the object-oriented design and analysis
- Typically, you are going to learn something about
  - requirement engineering (UML)
  - design pattern (including information hiding, design principles,etc)

What You Would Learn

- Of course the object-oriented design and analysis
- Typically, you are going to learn something about
  - requirement engineering (UML)
  - design pattern (including information hiding, design principles,etc)
  - refactoring

What You Would Learn

- Of course the object-oriented design and analysis
- Typically, you are going to learn something about
  - requirement engineering (UML)
  - design pattern (including information hiding, design principles,etc)
  - refactoring
  - web app frameworks? testing?

Your Projects

- Each group picks one problem from a pool

Your Projects

- Each group picks one problem from a pool
- We created 6 projects for you. Yet you can work on your own if you want to.

Your Projects

- Each group picks one problem from a pool
- We created 6 projects for you. Yet you can work on your own if you want to.
- Come talk to me if you want to come up with your own ideas. DO EXPECT THAT YOUR IDEAS MIGHT BE WITH A LOWER STARTING SCORE.

The Project List

- Auto-script: Automatically translate script input to desired script type.
- Paper-rank: Evaluate academic papers by their citation ranks
- Coursitter: List the results of whether it is able to register a course or not with the corresponding reasons.
- Prof-alert: Search and make appointments with your interested professors
- SUSTech-Monopoly: Build Monopoly game with SUSTech as the background
- SUSTeCards: Build card game with SUSTech as the background

Auto-script

- Build a website that automatically translate the input format to real scripts that are ready to execute.
  - The website provides a platform for executing command.
  - There are two kinds of user. The one is script designer and the other is script user.
  - The designer need to select one kind of scripts and upload their own scripts to server, and he also need to provide parameters for users
  - The user pass the parameters by the website and then the server execute the script according to the specific parameters, and finally it returns expectant response for user.
- Bonus hits
  - Try to design the website for as many scirpt types as possible.
  - ...

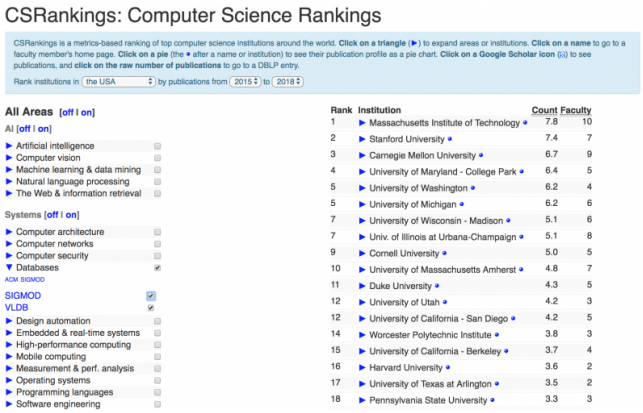
Paper-rank

IEEE Xplore® Digital Library

ACM DL DIGITAL LIBRARY

Google Scholar

中国计算机学会 CCF



CSRankings: Computer Science Rankings

CSRankings is a metrics-based ranking of top computer science institutions around the world. **Click on a triangle (▶)** to expand areas or institutions. **Click on a name** to go to a faculty member's home page. **Click on a pie (the ● after a name or institution)** to see their publication profile as a pie chart. **Click on a Google Scholar icon (🎓)** to see publications, and **click on the raw number of publications** to go to a DBLP entry.

Rank institutions in the USA by publications from 2015 to 2018

All Areas [off | on]

AI [off | on]
- ▶ Artificial intelligence
- ▶ Computer vision
- ▶ Machine learning & data mining
- ▶ Natural language processing
- ▶ The Web & information retrieval

Systems [off | on]
- ▶ Computer architecture
- ▶ Computer networks
- ▶ Computer security
- ▼ Databases

ACM SIGMOD

SIGMOD ☑
VLDB ☑

- ▶ Design automation
- ▶ Embedded & real-time systems
- ▶ High-performance computing
- ▶ Mobile computing
- ▶ Measurement & perf. analysis
- ▶ Operating systems
- ▶ Programming languages
- ▶ Software engineering

Rank	Institution	Count	Faculty
1	▶ Massachusetts Institute of Technology ●	7.8	10
2	▶ Stanford University ●	7.4	7
3	▶ Carnegie Mellon University ●	6.7	9
4	▶ University of Maryland - College Park ●	6.4	5
5	▶ University of Washington ●	6.2	4
5	▶ University of Michigan ●	6.2	6
7	▶ University of Wisconsin - Madison ●	5.1	6
7	▶ Univ. of Illinois at Urbana-Champaign ●	5.1	8
9	▶ Cornell University ●	5.0	5
10	▶ University of Massachusetts Amherst ●	4.8	7
11	▶ Duke University ●	4.3	5
12	▶ University of Utah ●	4.2	3
12	▶ University of California - San Diego ●	4.2	5
14	▶ Worcester Polytechnic Institute ●	3.8	3
15	▶ University of California - Berkeley ●	3.7	4
16	▶ Harvard University ●	3.6	2
17	▶ University of Texas at Arlington ●	3.5	2
18	▶ Pennsylvania State University ●	3.3	3

Paper-rank

- Build a website that can evaluate the paper impact with new dimensions
  - Given a request with a paper's title, the website needs to find the full information of that paper (through **google scholar, ieee xplore, acm library**, etc). You can tell the coarse-grained ranking of that paper accoring to the its published venue (through **CCF ranking**, **CSrankings**, etc).
  - After extracting the citations of that paper, the website is able to identify the rank of each citation (through CCF ranking, CSrankings, etc).
  - You need to combine the two derived information to design metric(s) to evaluate the fine-grained impact of that paper.
- Bonus hits:
  - Add more functions, such as ranking of all the papers inside of/cross domains within certain amount of time.
  - Consider the comments of that paper in its citations. More positive comments should lead to higher ranks.
  - ...

Coursitter

- Build a website that can demonstrate the results of registering the requested course on one's own.
  - Given a requested course from one student, the website should be able to deliver whether the student can register this course and the corresponding reasons.
  - There might be various reasons why student cannot register the course, e.g., inadequate prerequisite coursework, cross-year registering, academic integrity issue, etc.
- Bonus hits:
  - try to figure as many reasons as possible
  - try to propose candidate courses for student if he/she fails to register the requested one.
  - ...

Prof-alert

- Build a website that helps student find the professors with the research interests he/she is interested in and make appointments with them.
  - Given the input keywords regarding research interests or topics, e.g., “autonomous driving” or “software engineering”, the website can return a list of professors relevant to these keywords.
  - The student can select any professor and make appointment with him/her. Specifically, you need to design a calander, where the professor can fill the time slots in advance and the student can fill any of the rest to automatically make appointments with professors.
- Bonus hits
  - try to make a search engine with full consideration of search scale.
  - try to strengthen the user experience of the calander.
  - ...

SUSTech-Monopoly

- Monopoly

SUSTech-Monopoly

- SUSTech

SUSTech-Monopoly

- Build a Monopoly game with SUSTech as background
  - Replace hotels, jails, hospitals, etc. with your teaching buildings, labs, cafes, sportsfield, etc.
  - Replace winning/losing money with your success/failure of learning courses, publishing papers, attending competitions, etc.
  - Replace thieves, police, etc. with professors, staffs, etc.
  - Winning decisions are made according to your performace during college. You can design your rules of upgrading/downgrading grades of four years.
- Bonus hits
  - Design your game policy as funny and reasonable as possible
  - Fancy GUIs are welcome
  - ...

SUSTeCards

- Legends of Three Kingdoms

SUSTeCards

- Build a card game with SUSTech as background
  - Replace master, loyalists, rebels with professors, classmates, staffs, etc.
  - Replace the rules with your interactions with professors, classmates, staffs, etc.
- Bonus hits
  - Design your game policy as funny and reasonable as possible
  - Fancy GUIs are welcome
  - ...

Getting Started

- Lab Sections
  - Meet “your” TA
  - Form a project group (5)
  - Do the first lab tutorial
- Be ready for class
  - Come with your own machine if at all possible. With IDE (recommended)

A LITTLE SOMETHING ABOUT YOU

QUESTIONS?