2018 COMPUTER SCIENCE FRESHMAN BOOTCAMP

Facilitated by
  CS CS Profs Patty Kraft and Jo Ann Lane
  Several CS Upperclassmen

BOOTCAMP SCHEDULE

- 1:00-1:15   Introduction and Overview (including Ice Breaker #1)
- 1:15-2:20   Computer Science "big picture"
- 2:25-2:50   Suggestions to better manage your CS program outcomes
- 2:50-3:15   Ice Breaker #2 – get to know each other

ICE BREAKER #1: 10 MINUTES

- Break into groups of 3-4 and introduce yourselves
  - Name
  - Where you are from
  - Why you decided to major in computer science
  - Dream job after college
  - Favorite way to unwind

WHAT MIGHT A COMPUTER SCIENCE MAJOR DO ON A PROJECT?

- Software engineering
  - Architecture
  - Security
  - Data
- Software development/ programming
  - Applications
  - System software
  - Database development
  - User interfaces

- Development environment
  - System administration
  - Network administration
- Support engineering
  - Hardware engineering/embedded software
  - Security engineering
  - Quality assurance/testing
  - Configuration management
  - User training
  - Technical writing
- Leadership roles/management

EXAMPLE CAREER CHOICES (MAY BE MORE THAN ONE)

- Information technology
  - Healthcare
  - Business systems (accounting, payroll, etc.)
- Education
- Big data/data mining
- Web development
- Government
  - Federal
  - State
  - Local

- Security
- Embedded systems
  - Automotive
  - Aircraft
  - Communications (e.g. cellphones)
  - Healthcare devices
- Systems
  - Operating systems
  - Development tools (e.g., Windows, Linux, Apple)
- Services
  - System administrator
  - Network administrator

Might consider a minor even though one is not required!

A QUICK VIEW INTO A DAY IN THE LIFE OF AN INDUSTRY SOFTWARE TEAM

- Managers
- Senior engineers
- Mid-level engineers
- Entry-level engineers/interns

SDSU CS PROGRAM EDUCATIONAL OBJECTIVES

- Within a few years of graduation, SDSU Computer Science Bachelor of Science Degree graduates are expected to effectively integrate and apply in one or more fields
  - Key principles and practices of computing
  - Writing and verbal communication skills to effectively interact with clients, management, and team members
- With these skills, students graduating with a B. S. in Computer Science will have the requisite background to support future goals with respect to employability and graduate school:
  - Employability: Graduates will be immediately employable in high-technology companies that utilize their computing education
  - Graduate School Preparedness: Strong graduates will be prepared to enter graduate programs in Computer Science.

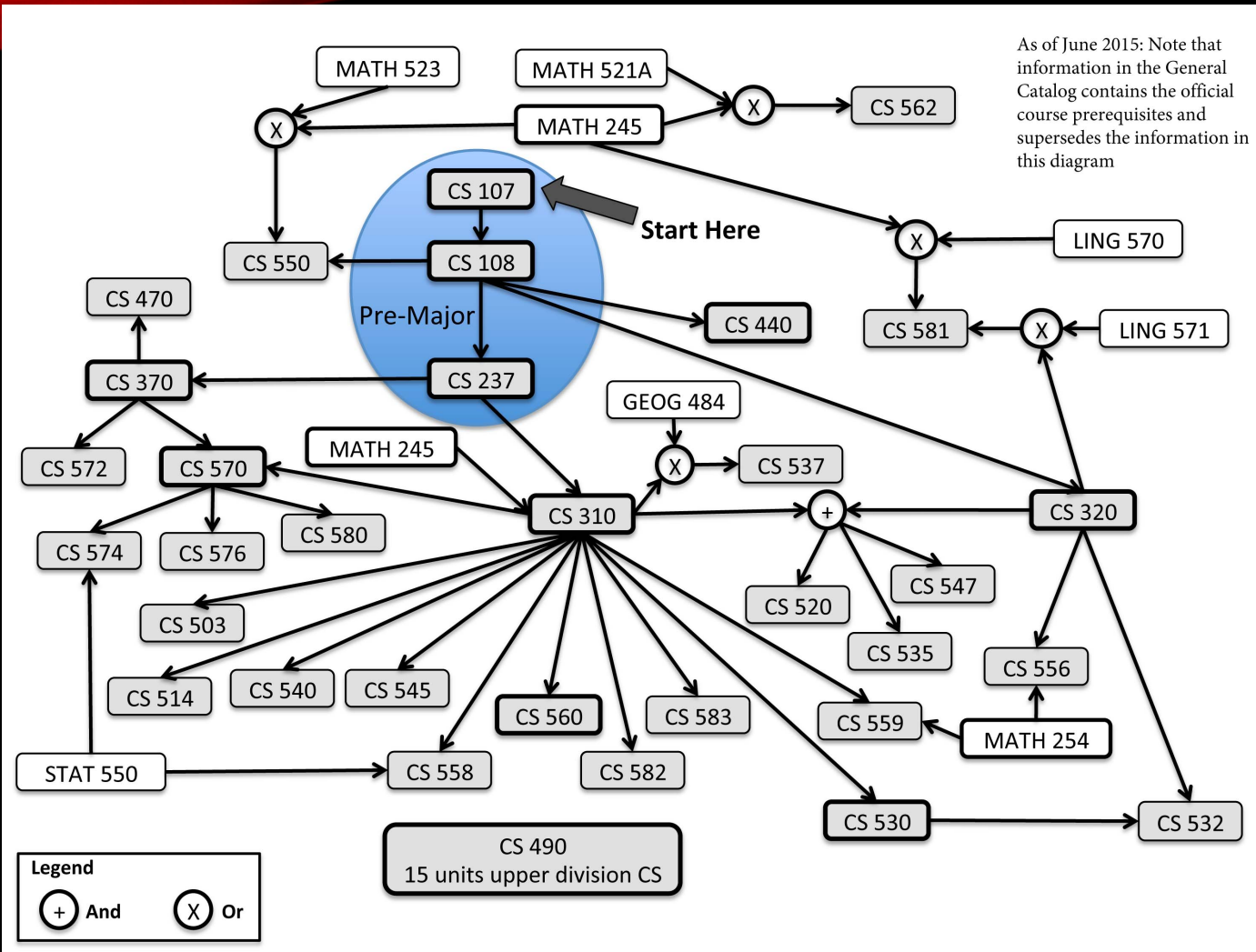# SDSU CS PROGRAM EXPECTED OUTCOMES

Upon successful completion of the SDSU CS BS Degree, students will be able to demonstrate the ability to:

- Apply knowledge of computing and mathematics appropriate to the discipline

- Analyze a problem, and identify and define the computing requirements appropriate to its solution

- Design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

- Function effectively on teams to accomplish a common goal

- Honor professional, ethical, legal, security and social issues and responsibilities

- Communicate effectively with a range of audiences

- Analyze the local and global impact of computing on individuals, organizations, and society

- Engage in continuing professional development

- Use current techniques, skills, and tools necessary for computing practice.

- Apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

- Apply design and development principles in the construction of software systems of varying complexity.

HOW TO BE SUCCESSFUL IN YOUR CS STUDIES

- Build strong programming and software architecting skills
  - Attend classes and labs
  - Focus on examples and take notes
  - Interact with Teaching Assistants (TAs) and instructor
  - Use the internet for additional details/tutorials
- Learn to collaborate and work with other people/classmates on larger projects
  - Take advantage of study groups if allowed
  - Spend time working with teammates when team projects are required
  - Key to successful teams: Plan the work and work the plan
- Follow
  - *SDSU Student Rights and Responsibilities:* *http://go.sdsu.edu/student_affairs/srr/Default.aspx*
  - *ACM/IEEE CS Ethics Credo (copy at end of slides)*

RELATIONSHIPS BETWEEN CS REQUIRED COURSES

As of June 2015: Note that information in the General Catalog contains the official course prerequisites and supersedes the information in this diagram

Start Here

Most current version can be found at http://www.cs.sdsu.edu/degree-requirements/

HOW TO SELECT ELECTIVES OR A MINOR

- Depends on what you plan to do with your CS degree
  - Where you want to work
    - Examples: Microsoft, Facebook, Amazon, aerospace/defense contractor, healthcare organization, government organization
  - What types of software you want to work with
    - Examples: Web-based software, large user application, system software, software embedded in devices, or "everything"
  - What role you prefer on a software project
    - Examples: software architect, programmer, tester, configuration manager, quality assurance/tester
- Why one might want to consider a minor (even though one is not required)
  - If you have a strong desire to work in a given organization or in a given domain, you may want a complementary minor

WHAT MAKES A GOOD PROGRAMMER

- Develop strong programming and computer science skills
- Stay current with new advances in technology that pertain to your areas of expertise (or areas where you would like to develop expertise)
- Be collaborative with team members
  - Volunteer in your strength areas
  - Volunteer to investigate new technologies/tools
  - Share information/help train rest of the team
- Be inviting to new members on a project
  - Show them where to find information about the project
  - Help them understand the development environment tools/libraries
  - Introduce them to other key project members/organization experts

HOW TO GET HELP WHEN YOU ARE FEELING LOST/CONFUSED

- Lost/Confused
  - TAs
  - Faculty
  - CS advisors
  - Other students

- And if you are just feeling lonely or wanting to connect with other CS students outside of class, check out the CS clubs
  - SDSU CS Association for Computing Machinery (ACM) Club (https://acm.sdsu.edu/)
  - Mechatronics (http://www.cs.sdsu.edu/mechatronics-club/)
  - Cyber Defense Team (http://www.cs.sdsu.edu/cyber-defense-team/)
  - Artificial intelligence (https://aiclub.sdsu.edu/)
  - Game programming

CS 107 INTRODUCTION TO PROGRAMMING

CS 108 INTERMEDIATE PROGRAMMING

What can you expect? What was it like for former students?

Let's ask them!

GMCS BUILDING

5th floor: Professor offices

4th floor: Lab (Room 425)

        CS/Math Office (Room 413)

3rd floor: Classrooms

WORKING IN TEAMS: WHEN IT IS GOOD AND WHEN NOT TO

When in doubt, ASK!!!

- When working in teams is good
  - Some CS courses require team projects
  - Working within CS-related campus clubs and CS-related events
    - Mechatronics (http://www.cs.sdsu.edu/mechatronics-club/)
    - Cyber Defense Team (http://www.cs.sdsu.edu/cyber-defense-team/)
    - Artificial intelligence (https://aiclub.sdsu.edu/)
    - Game programming
    - SDSU CS Association for Computing Machinery (ACM) Club (https://acm.sdsu.edu/)
    - Various hack-a-thons that are announced throughout the year
  - Other tasks where you might want to work in teams
    - Studying for quizzes and exams
    - Learning to use new tools or libraries
- When NOT to work in a team
  - When projects/homework assignments/exams are explicitly identified as individual work
  - When a project/homework assignment/exam has NOT been identified as team work

USEFUL ON-LINE RESOURCES

- Computer accounts provide access to
  - Blackboard
  - Development tools
  - Software repository
  - Test tools
- Information
  - Software design strategies
  - How to resolve known problems with software tools/products
  - Open source software
  - "How to" tutorials (e.g., you-tube, step-by-step procedures, etc.)

- *REMEMBER TO ACKNOWLEDGE ANY OUTSIDE SOURCE MATERIALS YOU USE IN ASSIGNMENTS… AND ONLY USE WHEN ALLOWED!!!*

ICE BREAKER #2

- Break into a different group of 3-4 and introduce yourselves (different than the first ice breaker)
  - Name
  - Where you are from
  - Why you decided to major in computer science
  - Dream job after college
  - Favorite way to unwind
  - 3 things about yourself
    - 2 true, 1 false
    - See if rest in your group can figure out which is false

QUESTIONS????

ACM/IEEE SOFTWARE ENGINEERING ETHICS CREDO

Software engineers shall commit themselves to making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession. In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the following Eight Principles:

1. PUBLIC – Software engineers shall act consistently with the public interest.

2. CLIENT AND EMPLOYER – Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.

3. PRODUCT – Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

4. JUDGMENT – Software engineers shall maintain integrity and independence in their professional judgment.

5. MANAGEMENT – Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.

6. PROFESSION – Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

7. COLLEAGUES – Software engineers shall be fair to and supportive of their colleagues.

8. SELF – Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession