

CSCE 240: Advanced Programming Techniques

Lecture 9: Object Oriented Concepts - Inheritance

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

6TH FEBRUARY 2024

Carolinian Creed: “I will practice personal and academic integrity.”

Credits: Some material reused with permission of Dr. Jeremy Lewis.
Others used as cited with thanks.

Organization of Lecture 9

- Introduction Section
 - Recap of Lecture 8
 - Announcements
- Main Section
 - Concept: Inheritance
 - Discussion: Home work #3 – due in Class 10
 - Discussion: Prog. Assignment #2 and Project discussion
- Concluding Section
 - About next lecture – Lecture 10
 - Ask me anything

Introduction Section

Recap of Lecture 8

- We rellooked at relationships between classes
- We discussed code organization – header and implementation files, when to separate
- We discussed programming assignment (PA) #1

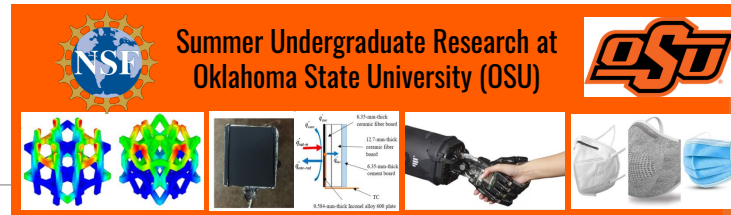
About Programming Languages in Course

- C++ is the main language for the course.
 - Used to demonstrate concepts and expect everyone to know it at the level that they can do peer evaluation and testing of each other's code in **home assignments**.
 - **For projects and programming assignments**, students have option to code in Java or Python as well.
- Cross-language understanding of concepts
 - Code in multiple languages is sometimes shown to demonstrate generality of concepts and specific peculiarities in implementation
 - UML diagrams will be used to convey cross-language concepts as well

In quizzes,

- **Questions will be about concepts, pseudo-code and UML diagram.**
- C++ code fragments may be shown or asked to be written, but they do not have to be running code. The quizzes will be in class and can be done on paper or a text editor like Google doc.

Summer Job Announcements



Program Details

The main goal of the National Science Foundation (NSF) funded Collaborative **Research Experience for Undergraduates (REU) site at Oklahoma State University (OSU)** is to increase undergraduate participation in research projects on technologies that lead to the development of Smart Personal Protective Equipment (SMAPP). The undergraduate students will have opportunities to work on cutting-edge research projects on material and structural design for protective equipment, heat flux measurement in fire conditions, human-robot interaction in dangerous environments and human perception on SMAPP.

Program Highlights

- \$6000 stipend for 10-week period
- Support for housing, transportation, meals, and round-trip travel costs to OSU, Stillwater, OK
- Travel to a national-level conference
- Professional development opportunities
- Field trips to local industries

Eligibility Requirements

- A US citizen or Permanent Resident
- GPA of 3.0 and above
- Must have at least one semester remaining to graduate
- Engineering / Social science majors
- Underrepresented groups and women are highly recommended to apply

Contacts

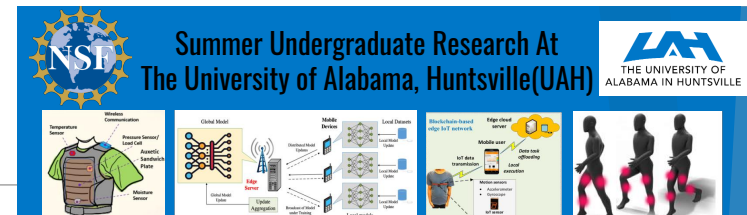
For more details, please email Dr. Haejun Park, Associate Professor, at haejun.park@okstate.edu or call on 405-744-9517

Venue and Dates

- OSU, Stillwater, OK
- May 20 – July 26, 2024 (10 weeks, Summer Program)

Application Materials

- Personal statement with research interest and plan for graduate studies
- CV / resume
- Transcripts
- Two professional references (requested through the application website)
- **Application deadline:** **March 22, 2024, 11:59 PM ET**
- Visit the webpage for more information <https://shorturl.at/cfuS9> or scan the QR code



The goal of the National Science Foundation (NSF) funded Collaborative **Research Experience for Undergraduates (REU) site at the University of Alabama in Huntsville** is to increase undergraduate participation in research projects on technologies that lead to the development of Smart Personal Protective Equipment (SMAPP). The undergraduate students will have opportunities to work on cutting-edge research projects on wireless sensor development, electronic circuit design, and artificial intelligence for human health monitoring.

Program Highlights

- \$6000 stipend for 10-week period (May 20 – July 26, 2024)
- Support for housing, transportation, meals, and round-trip travel cost to UAH
- Travel to a national-level conference for selected students
- Professional development opportunities
- Field trips to local industries

Eligibility Requirements

- A US citizen or Permanent Resident
- GPA of 3.0 and above
- Must have at least one semester remaining to graduate
- Engineering major
- Underrepresented groups and women are highly recommended to apply

Contact Details

For more details, please contact Avimanyu Sahoo, Assistant Professor at avimanyu.sahoo@uah.edu or call on 256-824-6307

Venue and Dates

- **Location:** UAH, Huntsville, AL
- **Duration:** May 20 – July 26, 2024 (10 weeks)

Application Materials and Deadline

- Personal statement with research interest and plan for graduate studies
- CV / resume
- Transcripts
- Two professional references (requested through the application website)
- **Application deadline:** **March 22, 2024, 11:59 PM ET**
- For the application process, please visit the website <https://shorturl.at/cfuS9> or scan the QR code



Main Section

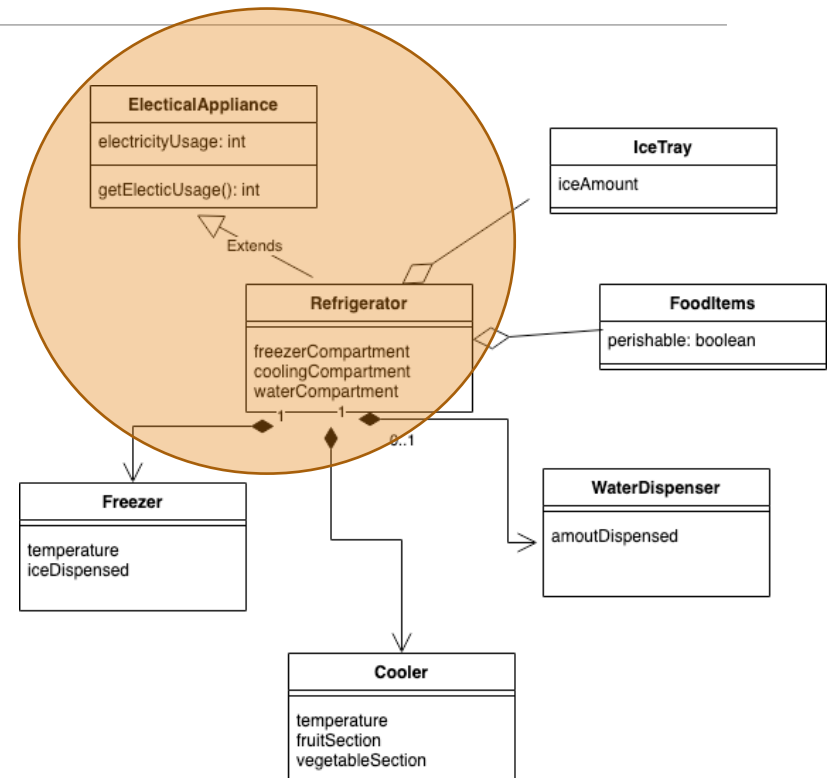
Concept: Inheritance

What is Inheritance ?

- A class “inheriting” or reusing **characteristics** from another, existing class
- Synonyms: subclassing, specialization, derived
- Analogy: child inheriting from a parent
 - “Course-CSCE-240” sub-class of “Course-Undergraduate”
 - “USA” specialization of “Country”
- What are characteristics
 - Data members
 - Enrollment, timing, syllabus: course domain
 - Capital, head-of-state, currency: country domain
 - Functions manipulating the data members

Why Use Inheritance ?

- Promote reuse
- Make code understandable, improve maintainability
- Promote security and data integrity
- Improve testing
- Improve code development productivity

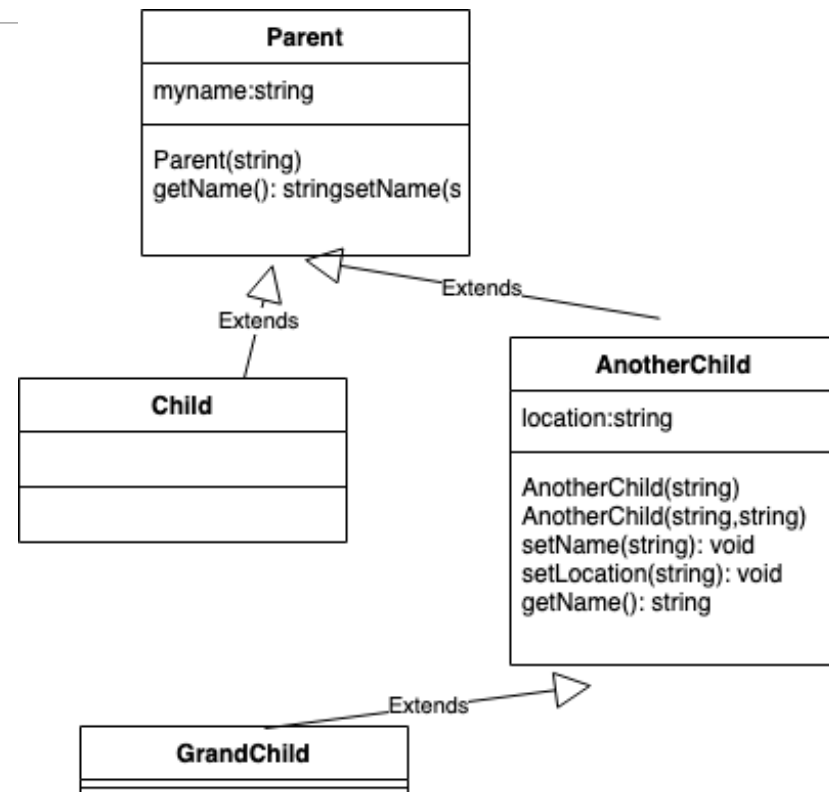


How to Use Inheritance ?

- Language independent syntax
- Illustration
 - 4 classes
 - 2 data members: myname, location
 - Access restrictions: private, protected, public

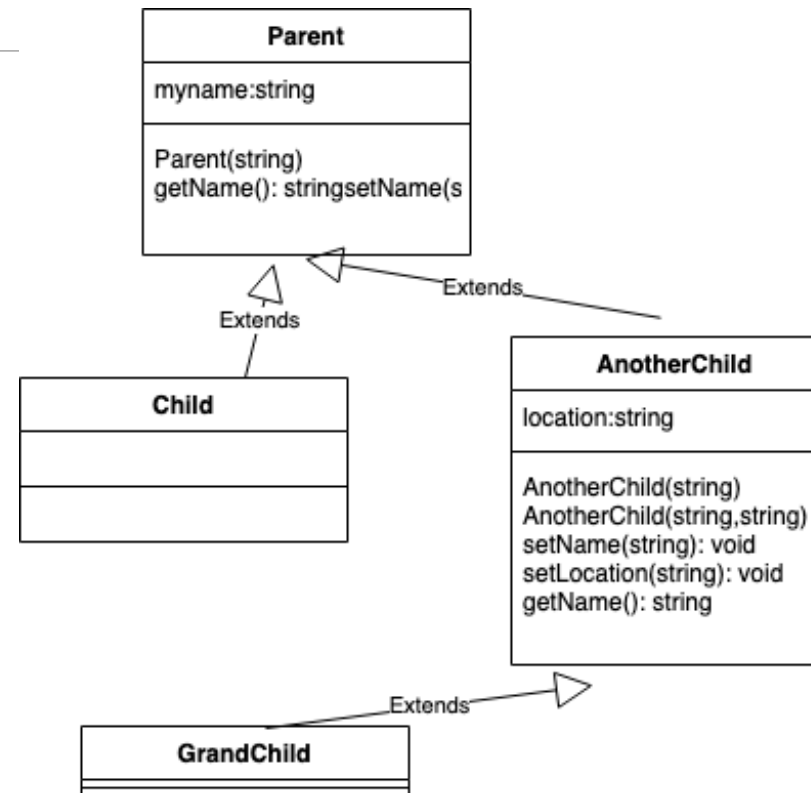
Code:

https://github.com/biplav-s/course-adv-proglang/tree/main/sample-code/CandC%2B%2B/Class9and10_C%2B%2B_OOAdv/src



Notes on Inheritance

- Code for classes Child and GrandChild are minimal
 - Code reuse happens by default
- A child can override the behavior of its parent



In-class Exercise

- Numbers and their operations
 - Operations: Addition, subtraction, division multiplication
 - Number types: natural numbers, whole numbers, rational numbers (fractions), irrational numbers, decimal numbers, binary, complex numbers
- Every one should take a number type and define its class

Home Work 3

Due Thursday, Feb 8, 2024

Programming Home Work (#3) – C++

Home Work #2

- Write a program called GeometricPropertyCalculator.
 - The program reads an input file (called input.txt). Each line in the file contains dimensions of a geometric shape – rectangle, shape and triangle. Specifically:
 - For rectangle, it contains – RECTANGLE <length-in-cm> <breadth-in-cm>
 - For circle, it contains – CIRCLE <radius-in-cm>
 - For triangle, it contains – TRIANGLE <side-1-in-cm> <side-2-in-cm> <side-3-in-cm>
 - The user specifies the property to calculate as argument to the program: 1 for AREA and 2 for PERIMETER
 - The program writes output lines to an output file (called output.txt) for each shape that it reads and the property – AREA or PERIMETER.
 - For example, for RECTANGLE and property as AREA, the program should write – RECTANGLE AREA <calculated value>
- Write GeometricPropertyCalculator in C++
 - It should support RECTANGLE, CIRCLE and TRIANGLE
 - It should support properties AREA and PERIMETER
 - If there is insufficient information, the program should give an error. E.g. TRIANGLE AREA “Not enough information to calculate”

Home Work #3

- Build a program called OOGeometricPropertyCalculator
 - Your new code will do the same as Home Work#2 but with OO design
 - It will have **4 classes: Shape – the parent, and its three children - Rectangle, Circle and Triangle**
 - Shape will have three members: **area, perimeter and errorMessage;** and at least **three functions getArea(), getPerimeter() and getErrorMessage().**
 - In your code, there will be a utility file (OOGeometricPropertyCalculator.cpp) with main() and will call the classes and functions. You can choose to have one or more files for the classes. (E.g, For the 4 classes, 4 headers + 4 .cpp files).
 - **You will also draw UML class diagrams for it**
- Functionality Reminder
 - The user specifies the property to calculate as argument to the program: 1 for AREA and 2 for PERIMETER
 - The program writes output lines to an output file (called output.txt) for each shape that it reads and the property – AREA or PERIMETER.

Programming Home Work (#3) – C++

- Code guidelines for the OO code you will write
 - Have sub-directories in your folder
 - src sub-folder, (or code) for code
 - data sub-folder, for input.txt and output.txt
 - doc sub-folder, for documentation on what the code does or sample output.
- In documentation
 - **Have a UML class diagram for the classes**
 - Observe how long was the code earlier and now. If you have to add a new functionality (like `getVertices()` to get all the vertices in a shape), how easy or hard will it be in HW2 code or HW3 code?

Discussion: Course Project

Course Project – Knowing About Companies

- **Project:** Develop collaborative assistants (chatbots) that offer useful information about companies
- Specifically, use the EDGAR dataset on companies at:
<https://www.sec.gov/edgar/searchedgar/companysearch>.
 - For Apple, it is: <https://www.sec.gov/edgar/browse/?CIK=320193&owner=exclude>
- **Each student will choose two companies (from thousand available).**
- Programming assignment programs will: (1) extract data about two companies from 10-k, (2) process it, (3) make content available in a command-line interface, (4) handle any user query and (5) report on interaction statistics.

Discussion: Nature and Simplifications

- Once you select a company, the scope of answers is fixed.
- Some simplifications
 - **Download local copy** v/s web query
 - **Read static content first**
 - **Handle a subset of content**
 - **Have default handling for questions** the chatbot does not understand
- Do project in a language you are most comfortable with
- Use all advanced programming concepts to simplify coding

Suggested Scope is a Drastic Simplification

- **Users:** 1
- **Modality:** text
- **Data:** static (optionally: dynamic – voting history)
- **Personalization:** none
- **Form:** command line
- **Purpose:** information provider
- **Domain:** specific to companies and their 10-K report

PA: Code Reviewing Rubric

- Look out for
 - Can one understand what the code is doing ?
 - Can one explain the code to someone else (non-coder) ?
 - Can one spot possible issues without running it?
 - Are the variables initialized ?
 - Are files closed?
 - Is their unnecessary code bloat ?
- What not to judge
 - Usage of language features, unless they are inappropriate

Assign rating (out of 100 -/+)

- -100: code not available
- -80: code with major issues (e.g., hardcopy path)
- -60: code with minor issues (e.g., uninitialized variables)
- -20:
- 0: (full marks): no issues
- +20: special features

PA: Code **Testing** Rubric Used

- Look out for
 - Does the program run as the coder wanted it to be (specification) ?
 - Does the program run as the instructor wanted it to be (requirement - customer) ?
 - Does the program terminate abruptly ?
 - Is there a hardcoding of directory ? Paths should be relative to code base directory.
 - Any special feature?
- What not to judge
 - Length of documentation. It can just be short and accurate.
 - Person writing the code

Assign rating (out of 100 -/+)

- -100: code not available
- -80: code with major issues (e.g., abnormal termination, incomplete features, infinite loop)
- -60: code with minor issues
- -20:
- (full marks): no issues
- +20: special features

Core Programs Needed for Project

- Prog 1: extract data from the district
- **Prog 2: process it (extracted data) based on questions**
- Prog 3: make content available in a command-line interface
- Prog 4: handle any user query and
- Prog 5: report statistics on interaction of a session, across session

Content Reference: 10-K Model

Concepts: 10-K, Parts, Items

Parts

- Part 1: Business Background and Risks
 - Item 1: Business
 - Item 2: Risk factors
 - Item 3: Properties
 - Item 4: Legal Proceedings
- Part 2: Operations and Disclosures
 - .. Market
 - .. Disclosures
- Part 3: Company Structure
 - Directors
 - Compensation
- Part 4: Financial Statements
 - Statements

Content Reference: Queries for (Answers) Data We Have

- What does the (company) do? // Answers in Part 1
 - What is the (company's) business?
 - What are (company's) risk factors?
 - What does (company) own?
 - ...
- Where does (company) operate? // Answers in Part 2
 - What has (company) disclosed?
- How is (company) structured? // Answers in Part 3
 - Who is (company's) CEO?
 - How much does (person) earn?
 - ...
- What was in (company) statements? // Answers in Part 4
 - ...

Concepts: 10-K, Parts, Items

Parts

- Part 1: Business Background and Risks
 - Item 1: Business
 - Item 2: Risk factors
 - Item 3: Properties
 - Item 4: Legal Proceedings
- Part 2: Operations and Disclosures
 - .. Market
 - .. Disclosures
- Part 3: Company Structure
 - Directors
 - Compensation
- Part 4: Financial Statements
 - Statements

Programming Assignment # 2

- Goal: **process extracted text based on questions**
 - Language of choice: Any from the three (C++, Java, Python)
- Program should do the following:
 - Take input from a local file which has content obtained from Prog#1 (when **company** name given as input)
 - Given an information type as input, the program will return its content
 - Examples: what is company's risk factors? What does company's CEO earn?
 - Input type can be given as command line argument.
Examples:
 - `prog2processor -t "what are IBM's risk factors?"` // Tell about company
 - `prog2processor -t "all information"` // Get all info for a company
- For demonstrating that your program works, have a file called "test_output.txt" showing the set of supported commandline options and output in the doc folder.

Concepts: 10-K, Parts, Items

Parts

- Part 1: Business Background and Risks
 - Item 1: Business
 - Item 2: Risk factors
 - **Item 3: Properties**
 - **Item 4: Legal Proceedings**
- Part 2: Operations and Disclosures
 - .. Market
 - .. Disclosures
- Part 3: Company Structure
 - Directors
 - Compensation
- Part 4: Financial Statements
 - Statements

• Code organization

- Create a folder in your GitHub called "prog2-processor"
- Have sub-folders: src (or code), data, doc, test
- Write a 1-page report in ./doc sub-folder
- Send a confirmation that code is done to instructor and TA, and update Google sheet

Concluding Section

Lecture 9: Concluding Comments

- We looked at inheritance relationship among classes
- Home Work #3 – due Feb 8
- Prog. Assignment #2 - due Feb 15

About Next Lecture – Lecture 10

Lecture 10: Object Oriented - Polymorphism

- OO – Polymorphism.
- Regular expressions – needed for project 2

Feb 1 (Th)	Code org (C++)	Prog 1 - end
Feb 6 (Tu)	OO – inheritance	Prog 2 - start
Feb 8 (Th)	Regex, OO - polymorphism	HW 3 due
Feb 13 (Tu)	Exceptions	
Feb 15 (Th)	Review: inheritance, Polymorphism	Prog 2 – end