Guided Exploration M01: Programming Fundamentals and Tools

**Points:** 25 (See Rubric in Canvas)

**Due Date:**  Due date listed in Canvas but some sections will be due as class participation before. Create Calendar reminders

* Final submission will be accepted up to 24 hours after the due date with a 10% penalty. Meaning if you turn it in at 12:01 am of the next day you will be deducted 10% of the total points from your score.
* If the assignment is more than 24 hours late, it will be a 0.

**Submission:** Upload files separately and do not upload a zip file.

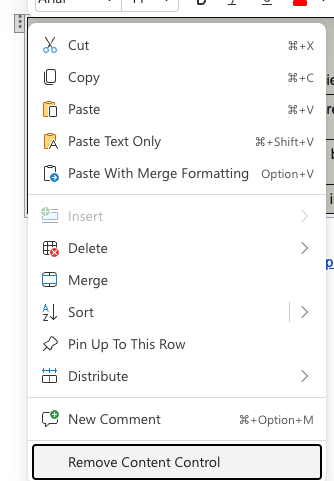
* **Three Document Files as PDF or Word document:**
  + GE01 Two Documents with your answers in the highlighted areas unless otherwise stated
    - Intro and Part 1
    - Part 2 and 3
  + Technical Document
* **1 Java File (. java files NOT CLASS FILES)**

**Objectives:**

* Analyze and apply fundamental object-oriented principles (encapsulation, inheritance, polymorphism, abstraction).
* Utilize arrays, classes, and objects to design structured programs.
* Develop effective technical documentation with examples, visuals, and clear explanations.
* Apply version control and debugging practices using Eclipse and GitHub.

**Effort:**  You are encouraged to collaborate to discuss concepts and explore writing code together. Write your technical documentation and answer the questions in your own words. You can use AI tools but follow [CS Academic Integrity and AI Usage Policy - Harding](https://docs.google.com/document/d/1SY4-RMJ2B9GkEbTzmpRVSfKKsgQnYWYF/edit)

If you download this as a word document and you have any problems typing your answer in the tables you must click on 3 dots and remove content control.



|  |
| --- |
| [Part 0 Intro: Learning and Collaborating Reflection](#_heading=h.hn1tkqh4utii)  [0.1 Working on Teams](#_heading=h.v2x5iw77f4bp)  [0.2 Learning and AI](#_heading=h.jznuqtf713p6)  [Part 1: Explore and Explain](#_heading=h.rlxhl11mpuu8)  [Part 2 Analyze and Apply](#_heading=h.qa3dtmgnnltf)  [Method printAllBooks()](#_heading=h.tli1oaximfqo)  [Method displayCountPerShelf()](#_heading=h.2pfu7bu4o8jq)  [Method displayOldest()](#_heading=h.3czwp09j44s6)  [Challenge (optional)](#_heading=h.so2gxd862n8h)  [Part 3 Reflection](#_heading=h.jkeogug6pm5o)  [Sample Main Method and Output](#_heading=h.ah1q0foh6jvw)  [Sample Main method to test](#_heading=h.mnyhm69zl25f)  [Sample Output](#_heading=h.ud6j651qh3tu) |

# Part 0: Learning and Collaborating Reflection

The reality of computer science is that there are so many languages, technologies and methodologies available and it is constantly evolving with new ones. So the goal is not just to understand the current technologies but developing skills to learn any new technology.

## 0.1 Working on Teams

In this course, we’ll work together on in-class activities, labs, and discussions. Good collaboration isn’t just about dividing work — it’s about building understanding together. Read [10 Collaboration Skills Examples](https://www.themuse.com/advice/collaboration-skills-examples) and answer the following

|  |
| --- |
| 1. Pick two of the 10 skills that you think are strengths you bring to the team and how you will bring those strengths to the team.   Problem solving and adaptability; With my background in military operations and high threat security management overseas, I have worked in a wide variety of environments, with an even broader spectrum of people. I am able to adapt to individual’s work and/or communication style’s within a team to find how to best work with them and produce results. |
| 2 Pick two of the 10 skills you think are most difficult for you and something you can do to try and improve in that area this semester.  Trust building and empathy are things I could work on. In my past lines of work, trust came pretty naturally and quickly, due to the environment we were in. So building trust with people outside of life and death situations can sometimes be a challenge. |

## 0.2 Learning and AI

Learning how to learn is an important component of this course as your career will be full of learning something new. O

|  |
| --- |
| 1 Read [Bluesky CEO Jay Graber warns: If you're a student, using AI means](https://timesofindia.indiatimes.com/technology/tech-news/bluesky-ceo-jay-graber-warns-if-youre-a-student-using-ai-means-/articleshow/122991750.cms) where Bluesky CEO’s Jay Graber warns that over-reliance on AI can impair critical thinking and long-term skill development and may contribute to “academic obsolescence” - declining ability to perform without AI  What skills do you risk losing if you depend on AI too early? |
| 2. [Navigating the AI Era: Self-Regulation](https://www.linkedin.com/pulse/navigating-ai-era-crucial-role-self-regulation-dr-ranjidha-rajan-4zl1c/) (Dr. Ranjidha Rajan, LinkedIn) Self-regulation and ethical responsibility are critical in the AI era — not just what you *can* do, but what you *should* do.  What does self-regulation look like when using AI for assignments? |
| 3 Summarize 3 Strategies you will use this semester to build skills, become a lifelong learner, and [Use AI Tools](https://docs.google.com/document/d/1SY4-RMJ2B9GkEbTzmpRVSfKKsgQnYWYF/edit#heading=h.awn48q1b3kpz) in a way that supports your own thinking. |

# Part 1: Explore and Explain

You will create a technical document to have as a resource for your projects, quizzes, future reference and even your resume. Here is a template to get you started if you want to use [Tech Doc CS2050](https://docs.google.com/document/d/1kY2IihTxB4QMDirJCcDuP5cpeDGhEsFC/edit)

When you add information in your own words demonstrating your understanding. Do not use exact words from lectures or images from lectures.

|  |
| --- |
| **DOs and DON’Ts** |
| DO: Provide clear and simple explanations in your own words (be concise)   * Use short sentences, small paragraphs * Use lists and tables to organize information * Use standard industry terms but explain in simple language * Use [AI Tools](https://docs.google.com/document/d/1SY4-RMJ2B9GkEbTzmpRVSfKKsgQnYWYF/edit#heading=h.awn48q1b3kpz) to help you learn but use lectures first   **DON’T copy my explanations from the slides as yours.** |
| DO Use Code Snippets and Visuals to reduce lengthy explanations:   * Include examples you explored or created * Include comments to describe parts of code * Include Screenshots such as from your debugger * Include your own drawings such as for memory, arrays, etc. * Include images you have AI create (cite that AI created) * Include images from the Internet with the link   **DON’T use my slides as your image and explanations** |
| DO Include Resources that give steps or help explain concepts   * You can include resources given in the lecture or find your own but   + Include website link with summary   + Include links to videos with summary of content   **DON’t**  use AI to create your technical documentation. |

**Topics to demonstrate in Tech Doc:**

**Analyze and Apply principles of computing to identify solutions**

* Review and apply prerequisite knowledge from CS1050:
  + Primitive and Non-Primitive Data Types
  + Methods and Stack Frames
  + Compare pass by value and pass by reference
  + Tracing program execution to understand scope and memory.
* Apply 1d and 2D arrays and Array of objects
* Explain how information is stored in memory (stack vs heap, references, arrays, objects).
* Define and implement **classes, objects, instance/static variables, and methods**.
* Apply **encapsulation** using access modifiers, constructors, and getters/setters.

**Apply Software Development Life Cycle to Design, Implement and Evaluate Solutions**

* Design classes using UML and use Test Driven Development to implement and unit test solutions.
* Write readable, maintainable code following industry coding standards (naming, indentation, comments).
* Create a program using an IDE with Debugger, version control (Git/GitHub), and Javadoc documentation.

# Part 2 Analyze and Apply

You will build the first iteration of a library app to manage a small library of books stored on shelves (2D array).

DO NOT WORRY IF YOU CAN’T COMPLETE THIS. You are not being graded for your code working. I am not going to run your code. I am looking to see that you did try writing the code on your own using concepts from the lectures.

Here are the [requirements (User Stories and Acceptance Criteria)](https://docs.google.com/presentation/d/1AiwAho2vGC3VN3p7Wid3r3i9O4JaWC9n/edit?slide=id.g348a0853a3f_0_51#slide=id.g348a0853a3f_0_51)

Implement and Test Using class design but feel free to add more methods to help keep code modular by following SRP and DRY

* [See sample main method and output below](#_heading=h.ah1q0foh6jvw)
* [Book Class Design](https://docs.google.com/presentation/d/1AiwAho2vGC3VN3p7Wid3r3i9O4JaWC9n/edit?slide=id.g348a0853a3f_2_88#slide=id.g348a0853a3f_2_88): Implement and Unit Test
* [Library Class Design](https://docs.google.com/presentation/d/1AiwAho2vGC3VN3p7Wid3r3i9O4JaWC9n/edit?slide=id.g348a0853a3f_2_106#slide=id.g348a0853a3f_2_106): Implement Library Class and Unit Test
  + class variables, constructor, getters/setters
  + Use TDD design, implement and test [addBook()](https://docs.google.com/presentation/d/1AiwAho2vGC3VN3p7Wid3r3i9O4JaWC9n/edit?slide=id.g348a0853a3f_2_255#slide=id.g348a0853a3f_2_255)
  + See below to use TDD to implement the following

Apply TDD for the following methods that includes

* UserStory
* List of acceptance criteria
* Fill out TDD table

## Method printAllBooks()

Output With books

|  |
| --- |
| ------------------------------------------------------------  All books in Test Library  Shelf Slot Book Details  ------------------------------------------------------------  1 1 "Unmasking AI" by Joy Buolamwini (2023)  1 2 "Hello World" by Hannah Fry (2018)  1 3 "Race After Technology" by Ruha Benjamin (2019)  1 4 "Deep Learning" by Ian Goodfellow (2016)  2 1 "Algorithms to Live By" by Brian Christian (2016)  2 2 "Weapons of Math Destruction" by Cathy O'Neil (2016)  2 3 "The Mythical Man-Month" by Fred Brooks (1975)  2 4 "Refactoring" by Martin Fowler (1999)  3 1 "The Pragmatic Programmer" by Andrew Hunt & David Thomas (1999)  3 2 "Peopleware" by Tom DeMarco & Tim Lister (1987)  3 3 "Computer Lib / Dream Machines" by Ted Nelson (1975)  3 4 "Extra Title" by Extra Author (2024)  (12 of 12 slots filled) |

Output without any books

|  |
| --- |
| ------------------------------------------------------------  All books in Test Library  Shelf Slot Book Details  ------------------------------------------------------------  (0 of 12 slots filled) |

User Story:

List Acceptance Criteria



Fill in table

|  |  |  |  |
| --- | --- | --- | --- |
| precondition | Action | Expected Result | Code Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Method displayCountPerShelf()

Output Examples

|  |
| --- |
| Shelf 1 has 0 books  Shelf 2 has 0 books  Shelf 3 has 0 books |
| Shelf 1 has 4 books  Shelf 2 has 4 books  Shelf 3 has 3 books |
| Shelf 1 has 4 books  Shelf 2 has 4 books  Shelf 3 has 4 books |

User Story:

List Acceptance Criteria



Fill in table

|  |  |  |  |
| --- | --- | --- | --- |
| precondition | Action | Expected Result (post condition) | Code Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Method displayOldest()

User Story:

List Acceptance Criteria



Hint: First create a method to convert a 2d array into a 1d array.

Fill in table

|  |  |  |  |
| --- | --- | --- | --- |
| precondition | Action | Expected Result | Code Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Challenge (optional)

This is for those that are more experienced and want a challenge.

* Add a static method in LibraryAppTestDriver class to read some of the data from a file.
* Add a library class method to print All Books To a File

# Part 3 Reflection

1. Testing & Debugging

|  |
| --- |
| 1.1 Give an example a problem you found when testing  1.2 How did writing tests first (or thinking about test cases) change the way you wrote your methods? |

2. Class Design

|  |
| --- |
| 2.1 How did using separate classes (Book and Library) make the program easier to understand and manage?  2.2 If all the code had been written inside main, what problems would that cause? |

3. Working with Arrays

|  |
| --- |
| 3.1 What challenges did you face using a 2D array for the shelves?  3.2 Explain how debugger helped working with a 2d array and include a screenshot of your 2d array for the bookshelves in memory in your IDE |

4. Working with Arrays

|  |
| --- |
| 4.1 What did you learn about the importance of planning and design before coding?  4.2 Which part of the program was the easiest for you to implement? Why?  4.3 Which part was the most challenging? What strategies helped you overcome it?  4.4 What are you most proud of from this assignment? |

# Sample Main Method and Output

## Sample Main method to test

|  |
| --- |
| **public** **static** **void** main(String[] args)  {  // --- unit test checks for Book ---  System.***out***.println("Unit Test Book Class");  Book unitTestBook = **new** Book("Unmasking AI", "Joy Buolamwini", 2023);  System.***out***.println("getTitle(): " + unitTestBook.getTitle());  System.***out***.println("getAuthor(): " + unitTestBook.getAuthor());  System.***out***.println("getYear(): " + unitTestBook.getYear());  System.***out***.println("stringOfBookDetails(): " + unitTestBook.stringOfBookDetails());  System.***out***.println();  System.***out***.println("Setting up Test Library");  **int** numberOfShelves = 3;  **int** shelfCapacity = 4;  System.***out***.println("Shelves (rows): " + numberOfShelves);  System.***out***.println("Slots per shelf (columns): " + shelfCapacity);  System.***out***.println("Total capacity: " + (numberOfShelves \* shelfCapacity));  System.***out***.println();  Library library = **new** Library("Test Library", numberOfShelves, shelfCapacity);  library.displayCountPerShelf();  library.printAllBooks();  library.displayOldest();  // Row 0  library.addBook(**null**);  library.addBook(**new** Book("Unmasking AI", "Joy Buolamwini", 2023));  library.addBook(**new** Book("Hello World", "Hannah Fry", 2018));  library.addBook(**new** Book("Race After Technology", "Ruha Benjamin", 2019));  library.addBook(**new** Book("Deep Learning", "Ian Goodfellow", 2016));  library.displayCountPerShelf();  library.printAllBooks();  library.displayOldest();  // Row 1  library.addBook(**new** Book("Algorithms to Live By", "Brian Christian", 2016));  library.addBook(**new** Book("Weapons of Math Destruction", "Cathy O'Neil", 2016));  library.addBook(**new** Book("The Mythical Man-Month", "Fred Brooks", 1975));  library.addBook(**new** Book("Refactoring", "Martin Fowler", 1999));  // Row 2  library.addBook(**new** Book("The Pragmatic Programmer", "Andrew Hunt & David Thomas", 1999));  library.addBook(**new** Book("Peopleware", "Tom DeMarco & Tim Lister", 1987));  library.addBook(**new** Book("Computer Lib / Dream Machines", "Ted Nelson", 1975));  library.displayCountPerShelf();  library.printAllBooks();  library.displayOldest();  System.***out***.println();  System.***out***.println("Test add more books than capacity...");  library.addBook(**new** Book("Extra Title", "Extra Author", 2024)); // should trigger "full" message  library.displayCountPerShelf();  library.printAllBooks();  library.displayOldest();  }// end main |

## Sample Output

Unit Test Book Class

getTitle(): Unmasking AI

getAuthor(): Joy Buolamwini

getYear(): 2023

stringOfBookDetails(): "Unmasking AI" by Joy Buolamwini (2023)

Setting up Test Library

Shelves (rows): 3

Slots per shelf (columns): 4

Total capacity: 12

Shelf 1 has 0 books

Shelf 2 has 0 books

Shelf 3 has 0 books

------------------------------------------------------------

All books in Test Library

Shelf Slot Book Details

------------------------------------------------------------

(0 of 12 slots filled)

Display Oldest: Library is empty.

Invalid book.

Added "Unmasking AI" by Joy Buolamwini (2023) at shelf 1, slot 1

Added "Hello World" by Hannah Fry (2018) at shelf 1, slot 2

Added "Race After Technology" by Ruha Benjamin (2019) at shelf 1, slot 3

Added "Deep Learning" by Ian Goodfellow (2016) at shelf 1, slot 4

Shelf 1 has 4 books

Shelf 2 has 0 books

Shelf 3 has 0 books

------------------------------------------------------------

All books in Test Library

Shelf Slot Book Details

------------------------------------------------------------

1 1 "Unmasking AI" by Joy Buolamwini (2023)

1 2 "Hello World" by Hannah Fry (2018)

1 3 "Race After Technology" by Ruha Benjamin (2019)

1 4 "Deep Learning" by Ian Goodfellow (2016)

(4 of 12 slots filled)

------------------------------------------------------------

Oldest books in Test Library

Earliest publication year: 2016

"Deep Learning" by Ian Goodfellow (2016)

Added "Algorithms to Live By" by Brian Christian (2016) at shelf 2, slot 1

Added "Weapons of Math Destruction" by Cathy O'Neil (2016) at shelf 2, slot 2

Added "The Mythical Man-Month" by Fred Brooks (1975) at shelf 2, slot 3

Added "Refactoring" by Martin Fowler (1999) at shelf 2, slot 4

Added "The Pragmatic Programmer" by Andrew Hunt & David Thomas (1999) at shelf 3, slot 1

Added "Peopleware" by Tom DeMarco & Tim Lister (1987) at shelf 3, slot 2

Added "Computer Lib / Dream Machines" by Ted Nelson (1975) at shelf 3, slot 3

Shelf 1 has 4 books

Shelf 2 has 4 books

Shelf 3 has 3 books

------------------------------------------------------------

All books in Test Library

Shelf Slot Book Details

------------------------------------------------------------

1 1 "Unmasking AI" by Joy Buolamwini (2023)

1 2 "Hello World" by Hannah Fry (2018)

1 3 "Race After Technology" by Ruha Benjamin (2019)

1 4 "Deep Learning" by Ian Goodfellow (2016)

2 1 "Algorithms to Live By" by Brian Christian (2016)

2 2 "Weapons of Math Destruction" by Cathy O'Neil (2016)

2 3 "The Mythical Man-Month" by Fred Brooks (1975)

2 4 "Refactoring" by Martin Fowler (1999)

3 1 "The Pragmatic Programmer" by Andrew Hunt & David Thomas (1999)

3 2 "Peopleware" by Tom DeMarco & Tim Lister (1987)

3 3 "Computer Lib / Dream Machines" by Ted Nelson (1975)

(11 of 12 slots filled)

------------------------------------------------------------

Oldest books in Test Library

Earliest publication year: 1975

"The Mythical Man-Month" by Fred Brooks (1975)

"Computer Lib / Dream Machines" by Ted Nelson (1975)

Test add more books than capacity...

Added "Extra Title" by Extra Author (2024) at shelf 3, slot 4

Shelf 1 has 4 books

Shelf 2 has 4 books

Shelf 3 has 4 books

------------------------------------------------------------

All books in Test Library

Shelf Slot Book Details

------------------------------------------------------------

1 1 "Unmasking AI" by Joy Buolamwini (2023)

1 2 "Hello World" by Hannah Fry (2018)

1 3 "Race After Technology" by Ruha Benjamin (2019)

1 4 "Deep Learning" by Ian Goodfellow (2016)

2 1 "Algorithms to Live By" by Brian Christian (2016)

2 2 "Weapons of Math Destruction" by Cathy O'Neil (2016)

2 3 "The Mythical Man-Month" by Fred Brooks (1975)

2 4 "Refactoring" by Martin Fowler (1999)

3 1 "The Pragmatic Programmer" by Andrew Hunt & David Thomas (1999)

3 2 "Peopleware" by Tom DeMarco & Tim Lister (1987)

3 3 "Computer Lib / Dream Machines" by Ted Nelson (1975)

3 4 "Extra Title" by Extra Author (2024)

(12 of 12 slots filled)

------------------------------------------------------------

Oldest books in Test Library

Earliest publication year: 1975

"The Mythical Man-Month" by Fred Brooks (1975)

"Computer Lib / Dream Machines" by Ted Nelson (1975)