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.

* **Two Document Files as PDF or Word document:** This document with your answers in the highlighted areas unless otherwise stated and Technical Document
* **Two Java Files (. java files NOT CLASS FILES):** Code from 2.1 Guess Number and 2.2 Calculate Grades

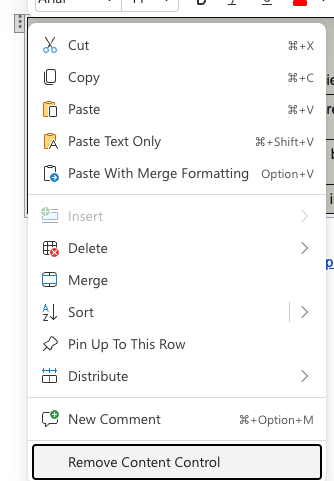
**Objectives:**

* Analyze and apply relevant theories, principles, and methods from computing, including algorithms, coding structures, and version control tools.
* Develop effective technical documentation to reflect on and solidify learning, with a focus on clarity and professional presentation.
* Apply version control and debugging practices using Eclipse and GitHub.

**Effort:**  You are encouraged to collaborate to discuss concepts and explore writing code together. Remember

* Review the [lectures](https://drive.google.com/drive/folders/1HTzIhv1tXvylB6tQF9fBIpjQBBY-pj_-) and links in the lectures to information first.
* Follow [CS Academic Integrity and AI Policy - Harding](https://docs.google.com/document/d/1SY4-RMJ2B9GkEbTzmpRVSfKKsgQnYWYF/edit)
* If you find you don’t understand the information in the lecture reach out to **Deb, Elaine or Heriberto**

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](#_am8t1cf4i3dj)  [0.1 Working on Teams](#_c253t059fegc)  [0.2 Learning How to Learn](#_imv5n52wsvc4)  [0.3 Hello World](#_g9m3fyvx8ckk)  [Part 1: Explore and Explain](#_4ddd3z5g8ss4)  [Part 2 Analyze and Apply](#_7m83xp254taj)  [2.1 Explore GuessNumber Code](#_2ktcry38aey)  [2.2 Calculate Grades](#_1t3h5sf)  [3 Reflection](#_sokkf1ov5gyk) |

# 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.   1. Adaptability. I can / will adapt and perform at the highest level within a small amount of time. I can use any software and learn any new path. 2. Active listening. If someone is speaking to me I will listen. It is very important to show civility in the classroom towards other students. |
| 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.   1. Communication. I am very reserved when it comes to showing my personal computer / projects. I need to work on communicating with others to improve my code / ideas. 2. Resilience. When something goes wrong I tend to get disinterested and leave from that situation. I need to be better at solving those issues and moving forward. |

## 0.2 Learning How to Learn

Learning how to learn is an important component of this course as your career will be full of learning something new. One of my favorite websites is [Train Ugly - How To Get More Out Of Your Practice](https://thelearnerlab.com/train-ugly/).

|  |
| --- |
| 1 Listen to this 30 minute podcast [Desirable Difficulties - The Learner Lab](https://thelearnerlab.com/desirable-difficulties/) and answer the following.  Explain desirable difficulties and how you think it relates to the learning in this class and in your future career.  Desirable difficulties can be defined as, what are the ways we can avoid getting into auto pilot.  I believe that avoiding auto pilots is important because, then you are not truly learning anything. Nothing is stimulated anymore. Some of your great ideas will not be used because you are not thinking anymore. |
| 2 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?  You will lose your critical thinking skills. The point is to make the student do something, even if it’s easy, so that they have to think for themselves. |
| 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.   1. Reviewing content, I can use a reader to help me understand the text without reading. 2. Generating helpful ideas for CSS or even some Python. 3. Reminding me about due dates and keeping me organized. |

## 0.3 Algorithms

 A picture containing diagram

Description automatically generated

"You can’t just build an algorithm, put it on a shelf and decide whether you think it’s good or bad completely in isolation. You have to think about how that algorithm actually integrates with the world that you’re embedding in." Hannah Fry

Read [Chapter 1](https://drive.google.com/file/d/1UNweLQrXpyFJ41N1lYy2ypCPp-9F41uu/view?usp=drive_link) from Hannah Fry, “Hello World, Being Human in the Age of Algorithms” and answer the following.

**Answer the following in your own words.**

|  |
| --- |
| 1. Describe an algorithm.  An algorithm is a set structure that explains to the computer what its task or script should be. E.g. the monkey, chair, banana problem. |
| 2. Summarize the four main categories of real world tasks that algorithms are doing.  Prioritization, classification, association, and filtering are the four main points. They work together in an algorithm to filter down items so that they remain specific to the user. |
| 3. Explain the two main approaches taken by algorithms. Include the pros and cons.  Rule-based algorithms: set by a human and the machine cannot interfere with these set instructions.  Machine-learning algorithms: Trials run by the machine to find the best algorithm for any solution. |
| 4. What can happen when we rely on machines to make decisions for us?  The machine will not think like a human therefore, the decisions will not be made for a human, these can be bad and lead us in a circle. |

# Part 1: Explore and Explain

**Purpose:** Professional communication to demonstrate your learnings by connecting different ideas from lectures or provided resources.

* Clearly explain key ideas in your own words and include supporting evidence such as code examples, images, screenshots, or charts.

You will create a technical document to have as a resource for your projects, quizzes, future reference and resume.

* [Tech Doc CS1050](https://docs.google.com/document/d/1zv2MW8HjNs8XzBHVNM3sjuLjkkIMs3-UE0rEefGaJpk/edit?tab=t.0): Download as word or create a copy of the google documentation.
  + Put in your git repository after you set up git and github.
  + Submit this as a word or pdf with your GE submission.
* Use headings to organize topics and create or update a table of contents to be able to quickly access information.
  + [Headings, Subheadings, and Table of Contents (Google Docs)](https://guides.lib.uni.edu/c.php?g=1243498&p=9758563)
  + [Headings, Subheadings, and Table of Contents (Microsoft Word)](https://guides.lib.uni.edu/c.php?g=1243498&p=9207893)

|  |
| --- |
| **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 code examples you created * Include comments to describe parts of your code * Include Screenshots, GIFs, Images   **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. |

# Part 2 Analyze and Apply

Demonstrates a clear analysis of a problem and applies relevant concepts effectively. Work shows independent thinking and understanding. You will use Git, GitHub and Eclipse to edit, version and back up your code.

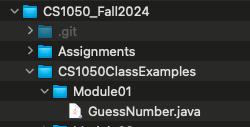
Resources to get started with Git/GitHub and Eclipse

* [Git and Github Version Control](https://docs.google.com/document/d/1mqMr_xE4cPg7IMeNPt0dBCIb7HCUp41SqKb_H-I45Sw/edit?tab=t.0)
* [Setup Eclipse](https://docs.google.com/document/d/1n2Ccp8WnDZuFas2bQuG6qnqFfR0f-KOk/edit)
* [Eclipse Cheat Sheet](https://docs.google.com/presentation/d/1zN6ux8dob_w-8bft0ey3BpuAKhaK4x8QmyowwRUbl0A/edit?slide=id.p1#slide=id.p1)

## 2.1 Explore GuessNumber Code

Be curious about the code as you explore.

1. Complete [Interactive Animation for Guess Number Program](https://liveexample-ppe.pearsoncmg.com/codeanimation/GuessNumber.html)
2. Go to your git repo and find the GuessNumber class example in Module01



1. Find the resource to add an existing Java file by dragging and dropping into your CS1050M01 Eclipse project.
2. Add comments to the code to help you understand. You can use parts of the code and comments in your technical document.
3. Version your guess number and technical document locally and then back up on your github cloud server.

|  |
| --- |
| 1. Take a screenshot showing you are using the debugger to show what is stored in the variable number and guess after you enter a guess. |
| 2. Describe 3 things you understand better about the code after using the debugger. |
| 3. Identify the warning in the code.  Show a screenshot of the warning in the program.  Find a resource to help fix the problem based on the warning message. You can use AI for help with syntax errors and warnings. |
| 4. Take a screenshot to show your GuessNumber.java file in your repo on the Github Cloud Server.  Explain why only the java file is showing in the repos and not the class file. |

## 2.2 Calculate Grades

You will create a project to store your assignment code and then create your program file. The program you create here will be updated in future guided explorations.

**Remember:** Commit and push your code often.

1. In your CS1050M01 project create a new class called **LastNameM01Calculations**
2. Apply best practices industry convention for naming variables, constants, methods and classes.
3. Update the code to add an initial comment. **This initial comment is required on all assignments**.
   1. Your name
   2. Class name
   3. Section (M/W) or (T/R) Due date
   4. Description

|  |
| --- |
| /\*  \* Name: [YOUR NAME]  \* Class: CS1050 (M/W or T/TH)  \* Description: Guided Exploration 01  \* This program will print a couple paragraphs about my passions and interests.  \* The program will calculate a final grade for this class based on the category weights  \*/  public class LastNameFirstNameGE01Calculation  {  {    }//main  }//Class GE01 |

1. **Explore** different methods to print out information.
2. **Displaying Output:** Add code to the **main** method to print 2 paragraphs about yourself. Some ideas: what you’re passionate about, what degree you’re working on, what you like to do, what are your goals etc. Only add a little bit of code at a time, run and compile to help you find errors if they occur.
3. **Working with Expressions:** Next add code to calculate your final grade based on the weights in the syllabus. Do not hardcode values in your expression. Use constants for percentage weights and variables for your grades.Remember to use correct naming conventions.
4. Display your name and final grade using printf to format the output to the console.
5. Add comments.
6. Commit your .java code file locally and push your code to github remote repository. Make sure you include comments.
7. Put a screenshot of your code in your github remote repository.
8. Use snippets of code, screenshots and explanations to answer the following in your technical document.
   1. Explain why this would not be a good way to implement your code to calculate a final grade for this class. double finalGrade = (.15 \* 95) + ( .2\* 92) + ( .25\* 88) + ( .2\*87 ) + (.2 \*93 )
   2. Errors
      1. Explain syntax error. Include a screenshot of your IDE showing there is a syntax error
      2. Explain logical errors. Explain what you can do to prevent logical errors.
      3. Explain runtime error.

# 3 Reflection

|  |
| --- |
| 1. Identify one specific challenge you faced in this exploration and what you did to overcome it. |
| 2. Team Collaboration  What is one strength of yours when working with your group in class?  What is one strength of your team?  What is one area you would like to improve about your team working together in class? |
| 3. Environment Set Up  On a scale of 1 to 5 rate yourself on using the following tools and include one thing you understand and one thing you need help on.  git/github rating:   * I understand how to * I need help on   If you are not using Eclipse include what IDE you are using  Eclipse rating:   * I understand how to * I need help on |
| 4. Now that you are submitting your first Guided Exploration   * What do you wish you would have asked for help on or what did you get help on? * How was your time management? Did you have to do a lot the day it was due? * What will you keep the same and what will you try to improve on when you do the next GE? |