Competency

In this project, you will demonstrate your mastery of the following competency:

- Translate requirements to solve problems computationally

Scenario

You work for a small company that creates text-based games. You have been asked to pitch an idea to your team for a text-based adventure game with a theme and environment of your choice. Your game must include different rooms, items, and a villain. The basic gameplay will require the player to move between different rooms to gather all of the items. A player wins the game by collecting all the items **before** encountering the villain. The player will have two options for commands in the game: moving to a different room, and getting an item from the room they are in. Movement between rooms happens in four simple directions: North, South, East, and West.

You must include the designs for your game as a part of your idea pitch. Specifically, you have been asked to provide a map that displays the different rooms and items. You have also been asked to use pseudocode or flowcharts to design code for moving between rooms and getting items. If your pitch gets approved, these designs will help your team members understand the pitch, and will help the team develop the game in the future.

Directions

In this project, you will break the problem down into a set of requirements for your game program. Then you will design your game by creating a storyboard and pseudocode or flowcharts. Remember, in Project One, you are only *designing* the game. You will actually develop the code for your game in Project Two.

1. Review the Sample Dragon Text Game Storyboard in the Supporting Materials section to see a sample storyboard for a dragon-themed game. You will begin by **creating a storyboard** to plan out your game. Using one of the templates located in the What to Submit section, write a short paragraph that **describes the theme of your game** by answering all of the following questions:
    - What is your theme? What is the basic **storyline**?
    - What **rooms** will you have? (Note: You need a minimum of eight.)
    - What **items** will you have? (Note: You need a minimum of six.)
    - Who is your **villain**?
2. Next, you will complete your **storyboard** by **designing a map that organizes the required elements of the game (rooms, items, and villain)**. Using the blank map in your template, organize the different rooms and the items in each room. The following requirements must be met:
    - There must be a minimum of *eight* rooms.
    - Each room must contain *one* item, with the exception of the "start" room and the room containing the villain.
    - The "start" room is where players will begin their moves and should not contain any items.
    - The room containing the villain should *not* contain any items.

    Remember, to win the game, the player must move through the rooms, collect all the items, and avoid the room with the villain until all of the items have been collected. Make sure that it is possible for the player to win the game. For example, the room with the villain should not block a room containing an item.

    Note: The blank map in the template is provided as a guide. You may add more rooms or change the locations of rooms to suit your needs. This map is for your planning purposes; the player will not have access to this map in the game. You will be able to use your map later when creating and testing your code as a part of Project Two.
3. Carefully review the Sample Dragon Text Game Walkthrough video and Sample Dragon Text Game Output reading, located in the Supporting Materials section. These will give you an understanding of how the text-based game should work. As you read, consider the following questions:
    - What are the different steps needed in this program? How might you outline them in a way that a computer can understand?
    - What information would you need from the player at each point (inputs)? What information would you output to the

player at each point?
    - When might it be a good idea to use “IF” and “IF ELSE” statements?
    - When might it be a good idea to use loops?
    - When might it be a good idea to use functions (optional)?

Note: You are not required to turn in anything for this step. However, this step is important to prepare you to design your code in Steps #4 and 5.

4. **Create pseudocode or a flowchart that logically outlines the steps that will allow the player to move between rooms** using commands to go North, South, East, and West. Use your notes from Step #3 to help you design this section of code. Be sure to address the following:
    - What **input** do you need from the player? How will you prompt the player for that input? How will you validate the input?
    - What should the program do if the player enters a valid direction? What **output** should result?
    - What should the program do if the player enters an invalid direction? What **output** should result?
    - How will you **control the program flow** with decision branching and loops?
5. **Create pseudocode or a flowchart that logically outlines the steps that will allow the player to get the item from the room they are in and add it to their inventory**. Use your notes from Step #3 to help you design this section of code. Be sure to address the following:
    - What **input** do you need from the player? How will you prompt the player for that input? How will you validate the input?
    - What should the program do if the player enters a valid item (the item in their current room)? What **output** should result?
    - What should the program do if the player enters an invalid item (an item not in their current room)? What **output** should result?
    - How will you **control the program flow** with decision branching or loops?

What to Submit

To complete this project, you must submit the following:

Design Document *or* Design Presentation
Submit your completed Design Document Template **or** Design Presentation Template, which should contain all of the designs for your program. Be sure that you have completed the following pieces of the template:

> **Storyboard (Theme Description and Map)**
> Include a paragraph (if using Word) or a slide (if using PowerPoint) that describes the theme, the basic storyline, the rooms, the items, and the villain. Submit your completed map with the layout of the different rooms and the items in each room. Your map should be on one page of the Word document or one slide of the PowerPoint presentation. You completed these items in Steps #1 and 2.
>
> **Pseudocode or Flowcharts**
> Include the pseudocode or flowcharts showing how the player will move between rooms and get the item from each room. Input, output, and the decision branching and loops that control the program flow should be clear. You completed these designs in Steps #4 and 5.

Supporting Materials

The following resources may help support your work on the project:

Reading: Sample Dragon Text Game Storyboard
This storyboard document includes a sample theme description and map for a dragon-themed text-based adventure game. Use this document as a guide for writing your own storyboard. Note: This storyboard does *not* include any pseudocode or flowcharts. You will need to include those elements in your final design document or presentation.

Video: Sample Dragon Text Game Walkthrough (8:24)

This video shows a sample dragon-themed text game. There is a brief description of the game, as well as a video that shows the game running and a player moving through different rooms and gathering items based on the commands. Review this video to help you understand how a text-based adventure game works. A video transcript is available: Transcript for Sample Dragon Text Game Walkthrough.

Reading: Sample Dragon Text Game Output
This document shows the sample inputs and outputs for the dragon-themed text game. Review the sample inputs and outputs to better understand how a text-based adventure game works. Use this reading to help you create the pseudocode or flowcharts for the sections of code.

Reading: A Mini History of Text Based Games
This optional reading will give you additional context about the history of text-based games.

Project One Rubric

Criteria	Exemplary (100%)	Proficient (85%)	Needs Improvement (55%)	Not Evident (0%)	Value
Storyboard: Theme and Map	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Creates a storyboard for a game by describing the theme and designing a map that organizes the required elements	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include description of theme, completion of map, or identification and organization of all required elements	Does not attempt criterion	20
Pseudocode or Flowchart: Logical Steps and Functionality	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or efficient manner	Creates pseudocode or flowcharts that logically outline the steps needed to meet the required functionality	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include logical ordering of steps or accounting for all the required functionality	Does not attempt criterion	30
Pseudocode or Flowchart: Input / Output	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Determines user inputs and outputs based on the given scenario	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include clear identification of inputs and outputs	Does not attempt criterion	20
Pseudocode or Flowchart: Program Flow	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or	Uses decision branching and loops to control program flow	Shows progress toward proficiency, but with errors or omissions; areas for	Does not attempt criterion	25

	efficient manner		improvement may include consideration of all paths the user can take through the program		
Articulation of Response	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Clearly conveys meaning with correct grammar, sentence structure, and spelling, demonstrating an understanding of audience and purpose	Shows progress toward proficiency, but with errors in grammar, sentence structure, and spelling, negatively impacting readability	Submission has critical errors in grammar, sentence structure, and spelling, preventing the demonstration of the understanding of ideas	5
				Total:	100%