Introduction to Engineering Design III

Choosing Team Projects

Today

Project Ideas

Recommendation: Pick something challenging that we can code in Python

Groups

- 2-3 person groups
- Each group must have a different project

Choosing a Project

- Example Python Project Ideas:
  - Tic Tac Toe - game
  - Tic Tac Toe – game + computer player
  - Anagram maker
  - Prime Factorization of a number
  - Hangman (modular, with functions)
  - Text analysis
- Or come up with your own idea!
- Think about how you might complete your project

Present Your Choice

1. Pick a project
2. Come up with a 1 sentence problem statement
   - "This system will allow one user to play the game hangman"
3. Explain your choice
   - Why did you choose this project?
   - How will you solve it?

Defining "Top Level Requirements"

- What are the things your system **must** do in order for it to work?

- Remember, you don't need to explain how to do it – just what it needs to do!
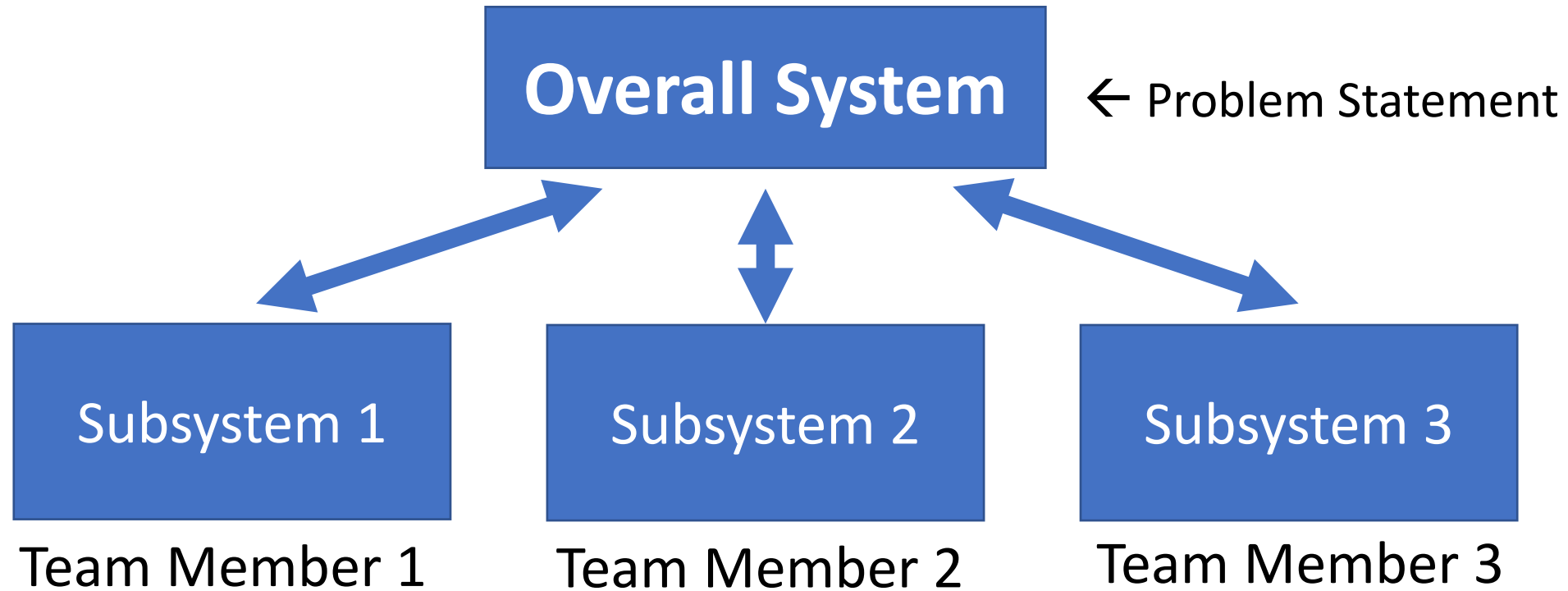
Define Top Level Requirements

Hangman Example:

1. The system shall have one player

2. The system shall have one secret word

3. The system shall have a specified number of turns

4. The player shall guess one letter each turn

5. The player wins when they have guessed all the letters in the secret word, and have not used all of their turns

6. The player loses when they used all of their turns, and the word has not been guessed

7. The system shall output information about the player's progress to the user

Define your top level requirements

- What are some things that your system **must** do in order for it to work?

- Remember, you don't need to explain how to do it – just what it needs to do!
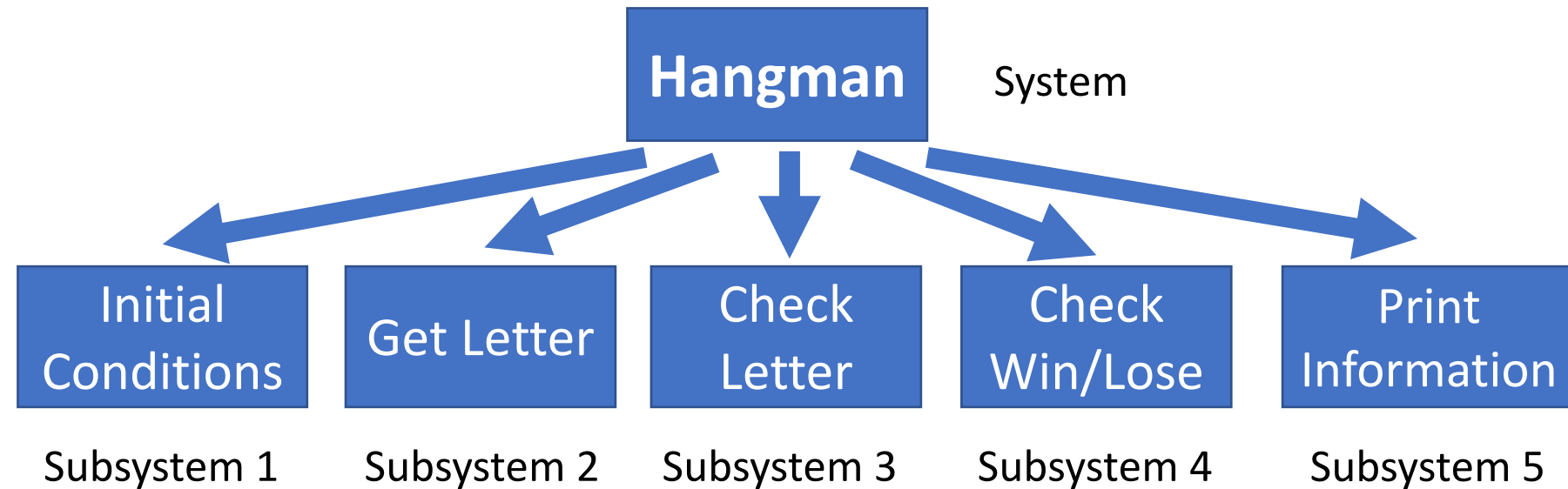
- Think of these as rules

Create a Project Structure

Create a Project Structure

- Based on your top level requirements, think of some ways you can group them together to make subsystems

Create Project Structure

Hangman **System**

```
                          [ Hangman ]
        ┌──────────┬─────────┼─────────┬──────────┐
        ▼          ▼         ▼         ▼          ▼
   [ Initial   ] [ Get   ] [ Check ] [ Check  ] [ Print      ]
   [ Conditions] [ Letter] [ Letter] [ Win/Lose] [ Information]

   Subsystem 1   Subsystem 2  Subsystem 3  Subsystem 4  Subsystem 5
```

Hangman Requirements:

1. The system shall have one player

2. The system shall have one secret word

3. The system shall have a specified number of turns

4. The player shall guess one letter each turn

5. The player wins when they have guessed all the letters in the secret word, and have not used all of their turns

6. The player loses when they used all of their turns, and the word has not been guessed

7. The system shall output information about the player's progress to the user

Define Subsystem Requirements
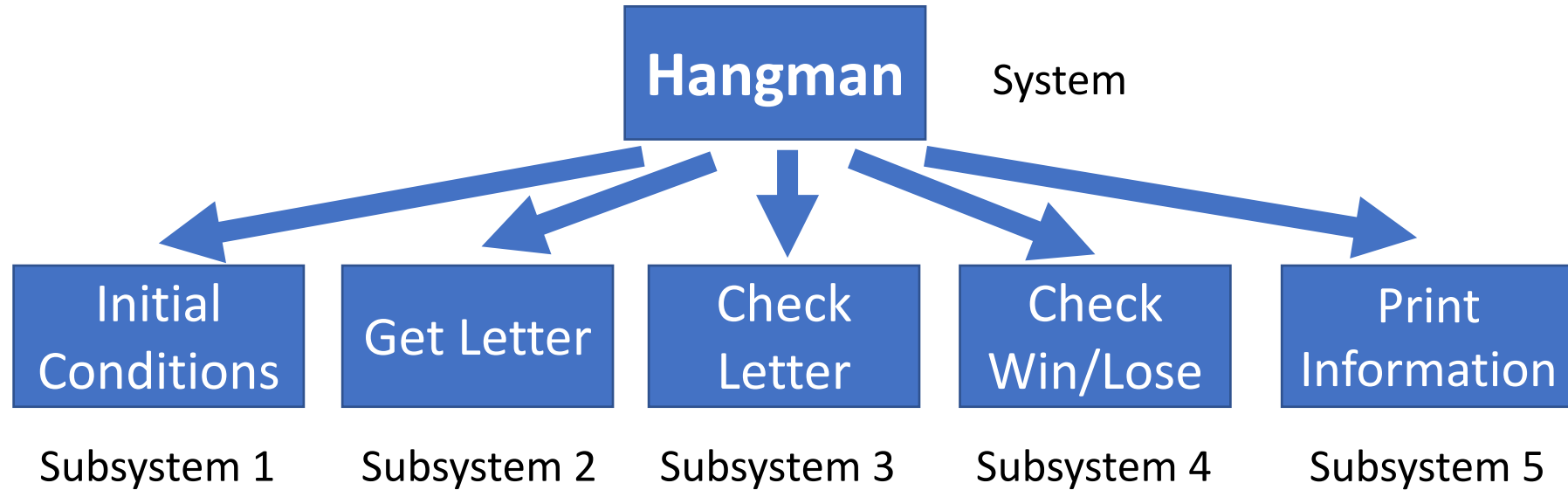
1. Initial Conditions
1.1 Subsystem shall get number of turns from the player and save as variable "turns_left"
1.2 Subsystem shall get secret word from the player and save as variable "secret_word"
1.3 Subsystem shall get name of the player and save as variable "player_name"

Preliminary Design Review - PDR

- Before we start building, we want to get feedback about our designs
- Create presentations to show what you are going to do

Prepare for Preliminary Design Review - PDR

1. What is your problem statement?

2. What are your top level requirements?

3. What is your project structure / what are your subsystems?

4. What are your subsystem requirements?

5. How are you going to build this?

6. Do you anticipate any problems building it?

We'll be presenting on Thursday / Friday