# Week 04: Instructor Notes

## Overview

Students will be summarizing Testing Techniques.

## Objectives

By the end of the week, each student will be able to:

* Summarize Testing Techniques (Exploratory, Domain, Flow, Box, etc.)

## Before the Week Begins (Prepare)

### Instructor Resources

Preparation:

Assign group presentation topics based on class size.

* Large Class (30-40)
  + Black box testing
  + White-box Testing
  + Unit testing
  + Integration and system testing
  + Acceptance and installation testing
  + Generating test cases
  + Partition testing, including equivalence classes and boundary analysis
  + Random Testing using uniform or operational distribution
  + Coverage to determine test adequacy
  + Regression testing
  + Usability Testing
  + Factory Acceptance testing
  + Verification and Validation testing
  + Usability testing
  + A/B comparative, Concurrent, Conformance
  + Performance/Stress, Scalability, Security
  + Alpha, Beta, Destructive, fuzz/gremlin
  + Formal Methods
* Medium Class (20-30)
  + Box: White and Black-box testing
  + Methodology: Unit, Integration, and System testing
  + Delivery: Acceptance and Installation testing
  + Generating test cases: Coverage, Partition and Random/Fuzzy testing
  + Scope: Regression and
  + UX: Usability
  + V&V:
    - A/B comparative, Concurrent, Conformance
    - Performance/Stress, Scalability, Security
    - Alpha, Beta, Destructive, fuzz/gremlin
* Small Class (7-20)
* Small Class (7-20)
  + Testing transparency: Black & White Box Testing, Performance, Scalability and Security

A screenshot of a computer

Description automatically generated with medium confidence

* + Testing Levels: Unit, Integration, System, and Regression, Acceptance

Timeline

Description automatically generated with medium confidence

* + Objective (Regression, Installation, Alpha/Beta Testing, Acceptance, Performance)

A picture containing graphical user interface

Description automatically generated

* + Generating tests: Partition, Boundary, Random, Code Coverage, Usability

Chart, treemap chart

Description automatically generated

* + Non-traditional testing: A/B comparative, Concurrent, Conformance, Alpha, Beta, Destructive, fuzz/gremlin

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generated

### Partnership/Group work:

This is a 4 part assignment. Prepare work is to research the testing technique. The teach is to prepare and present on the testing technique. They also need to submit 3-4 questions for the prove quiz, which you will enter.

## During the Week (Teach)

### Discussions:

* CSE270M04TeachTechniquesQuestions
  + Students will be posting questions for the Prove Quiz. Use these to create the quiz. Give 3-4 days to complete the quiz, even if it overlaps into the next module.
* CSE270M04TeachTechniquesPresentations
  + For students to answer questions in the Prove quiz, they must watch the corresponding presentations. This is where a student will post information about their presentation. Make sure the original presentation is available.
* CSE270M04TeachTechniquesCitations
  + In the past, some students have not included the citations, or the citations are found throughout the presentation. It is much easier for there to be one location and is graded accordingly.

## Looking Ahead

Next weeks announcements:

* Please create and post your Announcement for W05 by Friday of this week.

# Prepare

## Overview

There are several types of testing techniques, including exploratory, domain, flow, and box testing. Some techniques run throughout the lifecycle (last week), provide different approaches and coverage based on the software's need and purpose.

## Objectives

By the end of the week, each student will be able to:

* Summarize Testing Techniques (Exploratory, Domain, Flow, Box, etc.)

## Preparation Material

To be prepared for this module's activities, please read the following and be prepared to start or complete activities for Teach One Another and Prove assignments.

### Reading

* See [Reading Materials](../Reading/Reading.html)

Majority of your reading will be researching a Testing Technique.

* Please work with your partner from last week to research your assigned topic and present it.
* Review the teaching assignment, and Post your research for your team's assigned testing techniques, find different resources, and at least four need primary sources. [ [Please read BYU-I Library's Guide to sources, see Find Sources](https://libguides.byui.edu/cs/sources)]
* Create three questions about your topic. Use "[Creating Good Questions](file:///C:\Users\wac3\OneDrive%20-%20BYU-Idaho\GitHub\cse270-course\Course\CSE270GoodQuestions.html)" to create your questions.

# What questions do you have for your presentation?

## Overview

Develop good questions that students will use to test each other.

## Topics

You will need to come up with 3-4 Questions and upload them here so the Instructor can add them to the Prove Quiz at the end of the week.   
You and your partner will need to create questions such that the 2-3 question's values added together is 40 pts.   
You will be responsible for identifying the value of the question, or if not given, the default will be 2 pts for each question.

## Creating Good Questions

There have been some issues in the past with writing questions for the quizzes, and I would like to clarify some information.

### Question Difficulty and iLearn

iLearns 3.0 provide several different types of questions, and you have seen these in the quiz and other tests you have taken in other classes or tests:

* True/False (one correct answer or one incorrect answer) [auto grade]
* Multiple choice (one correct answer, m-incorrect answers) [auto grade]
* Multiple select (n-correct answers, m-incorrect answers) [auto grade]
* Matching (1-to-1 or n-to-1 or n-to-1+m; m-incorrect answers) [auto grade]
* Fill in the blank [Limited auto grade]
* Long-answers [No auto grade]

Break down of questions:

* True/False questions are the simplest type of questions.
* Most multiple-choice questions can be reduced down to a set of multiple T/F questions. Meaning if you have a multiple choice question and you don't know the answer, you can treat each possible choice as a T/F, where only one of them is True. NOTE: You are not allowed to have more than one throw-away answer, an answer that is not related to the question, i.e., My professor has crazy ties.
* This holds for Multiple-Select, except you can have multiple answers that are true.
* Matching says that all of the answers need to be correct and correspond to a subset of the answers provided (1-to-1 or n-to-1 or n-to-1+m). You can have the same amount of questions as answers (1-to-1). You can have more questions that match multiple times to the same answers (n-to-1). Or you have several questions to match only to a subset of the answers (n-to-1+m), you provide extra answers just to through off the asker.

### Auto-Grading

As you can see, the difficulty and complexity of the questions increase from T/F to Matching. Fill-in-the-Blank and Short/Long-answers are the next levels of difficulty, but there is a downside for online quizzes. They can't be auto-graded. So in order to allow multiple tries for this type of quiz and get instant feedback, I will not accept Fill-In-the-Blank, Sort, or Long answer questions. They can easily be converted to any of the previous types of questions.

## Fact-Based Questions vs. Comprehension or Reasoning

I am sure you have heard that Thoughts lead to Action, and Actions lead to Habits. Where do thoughts come from? They come from questions. Questions are smashing two ideas together. I am also sure that you know that there are good questions and not so good. The complexity of the question makes a huge difference in measurement, value, and quality. Here are a few categories of complexity questions.

* **Fact-Based Questions** are easy questions. Is it equal or not equal to the definition or situation. There is only one variable to deal with.
  + **True/False Example** (Statement of fact true or false): *The sun goes around the earth.*
  + **Multiple Choice/Select** (Definitions, Terms, etc.): *The command, in C++, to output text to the screen is: A. output; B. cout; C. echo; D. system.out.printf*
  + **Matching** (The examples here are incomplete, but the description gives you a general idea: *Match the terms with definitions...* or *Order the steps of the given process in the reading...*
* **Comprehension questions** are a bit more difficult, usually having to combine or compare situations. Comprehension questions can be like Fact-Based but explained in different ways in your own words. In a sense, one could say it comparisons or multi-variable for a given domain.
  + **True/False Example** (Using the comparison of facts to produce a conclusion): *(T/F) The earth travels faster around the sun than the spin of the sun.* or *(T/F) Loops are types of functions.*
  + **Multiple Choice/Select** (Comparing two different ideas, showing both the differences and similarities): *What is common between an if and a for loop: A. Initializing variable; B. Incrementing a variable; C. Comparison boolean; D. Executing a false code block*
  + **Matching** (Here everything is based on the reading): *Match the output of each line of code...* or *For each scenario given in the* ***reading****, match it with the term from the reading...* or *Order the steps in the scenario, based on the reading's given process...*
* **Reasoning questions** are more difficult because they follow a set of processes or equations. Reasoning questions are showing your work, or which of the following answers would be a solution to questions.
  + **True/False Example** (Providing logic reasoning to produce a conclusion): *The rotation of Mars with the rotation of Pluto, we can determine that the yearly earth rotation is shorter than Venus.* or *Functions can be used to perform looping*
  + **Multiple Choice/Select** (Providing possible solutions and choosing the best one): *Which looping statement would be best to use if you want to guarantee that the loop is executed at least once: A. Recursive call to the same function; B. A while statement; C. A do..while statement; D. A for loop*
  + **Matching** (Here, new ideas are compared to the reading): *For each scenario, (not from the reading), match them to the terms from the reading...* or *Order the steps of a given scenario, outside of the reading, based on the process provided by the reading...*

I know this is not a perfect comparison, but it helps illustrate the level of complexity.

Hints:

* If you can reframe the question to a lower question type, then it is that question type.
* Under certain conditions, question type can be the same, for example:
  + A multiple-choice question with two answers is the same as a True/False question
    - Is this question true:
      * A. Yes
      * B. False
    - T/F This is the equivalent
  + Multiple select questions with one right answer are the same as a multiple-choice question
    - This question only has one true answer, just like a multiple choice question
      * [] Correct
      * [] False
      * [] False
    - This question is the equivalent of a multiple-choice question
      * A. Correct
      * B. False
      * C. False
  + A matching with only two possible choices is the same as a multiple-selection.
    - This question has two possible matches (A. Correct or B. Incorrect)
      * 1. Matches with A
      * 2. Matches with A
      * 3. Matches with B
      * 4. Matches with B
    - This question is the equivalent of a multiple-select. Select the correct answers:
      * + [] This correctly matches
        + [] This correctly matches
        + [] This incorrectly matches
        + [] This incorrectly matches

## Submission

Upload a text document with your questions, use the following format:

* Question: <Provide the question>
* Answers:
  + <Possible Answer 1> : Correct or Incorrect
  + <Possible Answer 2> : Correct or Incorrect
* Type of Question: <Fact, Comprehension, Reasoning>

## Rubric

You need to provide the Instructor with your estimated question's points.   
Each question will be graded based on the matrix for a total of 40 pts.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Complexity** |  |  |
|  | ***Fact-Based*** | ***Comprehension*** | ***Reasoning*** |
| **Difficulty** | *(= - equal)* | *(compare)* | *(f(x))* |
| **T/F** | 2 pt | 4 pts | 6 pts |
| **Multiple-Choice** | 4 pts | 8 pts | 12 pts |
| **Multiple-Select** | 6 pts | 12 pts | 19 pts |
| **Matching (1-to-1)** | 8 pts | 16 pts | 24 pts |
| **Matching (n-to-1)** | 10 pts | 20 pts | 30 pts |
| **Matching(n-to-1+m)** | 12 pts | 24 pts | 36 pts |

**What citations did you use for your presentation on Testing Techniques?**

**Overview**

As part of the presentation on Testing Techniques, you need to provide ten at least sources.

**Topics**

Primary sources (4)

You will not find very much information on your topic in the course content. Instead, you will need to conduct research. Find enough quality sources to cover your topic. Your research needs to contain the following requirements:

* **Quality sources**: A quality source is a source in a peer-reviewed journal or written from a reputable source. Just about any source found through the BYU-Idaho library will qualify here. If you are unsure of the source's quality, look at the article's author and verify that it has sufficient credentials. Any source with an unknown author (such as Wikipedia) is not a quality source.
* **Cover the topic**: Support every fact you present in your research.

The best way to conduct this research is as follows:

1. Start at a primary and untrustworthy source like Wikipedia. This other type of source will give you an overview of the topic and rarely contains non-factual errors, but is not considered a primary source.
2. Look at the References and External Links section of the other source. These will often refer to a more trustworthy source you can cite, primary or secondary sources.
3. Find keywords and phrases in the primary source. Search in the Library or in [Google Scholar](http://scholar.google.com/schhp?hl=en) for quality sources using these keywords and phrases.
4. Continue this process until you have found quality primary sources on all aspects of the topic. You can expect to read a dozen or so articles until you settle on the half dozen or so you will need.

Post your research for your team's assigned testing techniques, find different resources, and at least four need primary sources. [ [Please read BYU-I Library's Guide to sources, see Find Sources](https://libguides.byui.edu/cs/sources)]  
The secondary source needs to come from each of the following categories. (Make sure that you indicate which resource type in your citation.)

* Institute, Organization, Community
* Research Papers
* Companies
* Books & Journals
* Use Cases, Case Studies, Published data
* Websites
* Video

**Rubric**

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Primary Sources 20%** | Provide 4+ sources | Provide at least 4 sources | Provide at least 2-3 sources | Provide at least 1 sources | No primary sources were identified or found. |
| **Secondary Sources 20%** | Provide 4+ sources | Provide at least 4 sources | Provide at least 2-3 sources | Provide at least 1 sources | No primary sources were identified or found. |
| **Other Sources 20%** | Provide 4+ sources | Provide at least 4 sources | Provide at least 2-3 sources | Provide at least 1 sources | No primary sources were identified or found. |

The distribution of points starts at 50 percent for the minimal participation expectation and additionally stated expectations increase the percents.

# Where is your group's presentation?

## Overview

Become familiar with the different testing techniques (breadth) and specialized in one (depth).

## Topics

### Research Your Topic

Your team will be assigned a Testing Technique (Exploratory, Domain, Flow, Box).

* Black-box testing
* White-box testing
* Unit testing
* Integration and system testing
* Acceptance and installation testing
* Generating test cases
* Partition testing, including equivalence classes and boundary analysis
* Random testing using uniform or operational distribution
* Coverage to determine test adequacy
* Regression testing
* Factory Acceptance testing
* Verification and Validation testing
* Usability testing
* A/B comparative, Concurrent, Conformance
* Performance/Stress, Scalability, Security
* Alpha, Beta, Destructive, fuzz/gremlin
* Formal Methods

Research (50 pts Total) your team's assigned testing techniques (5 pts for each category\* 10 citations) find ten different resources, at least one from each of the following categories. (Make sure that you indicate which resource type in your citation.) See [What citations did you use for your presentation on Testing Techniques?](file:///C:\Users\William\OneDrive%20-%20BYU-Idaho\GitHub\Home\byui-cse\cse270-course\Teach\CSE270M04TeachTechniquesCitations.html)

* Institute, Organization, Community
* Research Papers
* Companies
* Books & Journals
* Use Cases, Case Studies, Published data
* Websites
* Video

### Presentation

Create a presentation on your topic. Make sure you include the following information:

* Description
  + Describe the technique.
  + What are the advantages and disadvantages of using this technique?
  + What does a test look like for this technique?
* Relationship to class topics:
  + Which Lifecycle Phases (Module 1) does this technique used?
  + How does it relate to SWEBok Chap 4 Testing (Module 1)?
  + What Roles and Responsibilities are Needed?
* Relations to other techniques
  + What are the previous and post-testing techniques? Which testing technique needs to come before and after this testing technique?

### Create VIDEO

Create a video of your presentation. All members of your team must participate—post-presentation and link to the video.

### Questions:

Your team will have to create 2-3 questions. Please post them on the other Teach Discussion board. You will have points taken off if you post them here. See [What questions do you have for your presentation?](file:///C:\Users\William\OneDrive%20-%20BYU-Idaho\GitHub\Home\byui-cse\cse270-course\Teach\CSE270M04TeachTechniquesQuestions.html).

### Post

Post your presentations here.

## Rubric

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Slides 20%** | All the information is in a clear and easy-to-understand way | Professional presented. | All the required information is present. | The quality of the slides lacks in one or more ways. | The slides do not help present the information for the presentation, or there are no slides. |
| **Video Presentation 30%** | The presentation is engaging | All the information is professionally communicated | It is easy to understand the presentation | One or more aspect of the presentation distracts from the intended message | It is difficult to understand the presentation, or the presentation is missing information |
| **Description 10%** | Evidence complete understand of technique | Provided demonstration of technique. | Provided description, adv/disadv, and example | Provided description, adv/disadv, or example | Advantages nor Disadvantages provided |
| **Relation to Class Topics 20%** | Identified additional references outside of previous class topics | Identified additional references to previous class topics | Identified at all provided reference to previous class topics | Identified at least one reference to previous class topics | No reference to class topics. |
| **Pre/Post Techniques 20%** | Provide an example of a workflow. | When to use the technique is identified. | Identify pre and post techniques. | Identify pre/post techniques. | No mention of pre/post techniques. |

The distribution of points starts at 50 percent for the minimal participation expectation and additionally stated expectations increase the percents.

# Prove: Questions from the Presentations

## Overview

Enumerate and explain the key points on a variety of software testing topics

## Instruction

As soon as all the groups have submitted their topic questions, the Instructor will enter them into the Prove Quiz. After Prove Quiz is ready, the Instructor will make the Quiz available for five days.   
You will be required to review your classmates' presentations to answer the related questions.

## Submission

Make sure your team posts your presentation, and everyone's name is on it.  
Take Quiz