

# INSY 560 Test 2 Objectives

The following are things you should be able to do or discuss on the second test. The test will probably consist of approximately six essay- or form-completion type questions. You will be expected to write approximately 1-2 pages for each answer. Your answers will require both direct writing, as well as diagramming of the information. The form-completion type questions would provide information and one or more forms, and you would have to fill out the forms based on the information provided.

## General

- Describe / explain in detail the various *scripts, forms, logs, and standards* which are used in each portion of the PSP, and be able to fill out each type of form or log based on your actual work or if given sample data.

## Ch. 9: SW Quality Management

- Describe how *SW quality* is defined.
- Name, describe, and discuss the *2 aspects of SW quality*
- Explain why SW quality is an *economic* issue.
- List the *3 costs* of finding & fixing defects.
- Explain the *relative costs* of fixing defects in *various phases* of SW development.
- Explain *Brooks' rule of thumb* for when a project is half-way finished.
- Define & explain the concept of *yield* WRT SW quality management. Discuss *how much you can know about yield* at the various phases of SW development, & how it can be useful in planning, tracking, & managing a SW project.
- Define *COQ*, & name & explain its *3 components*
- Identify & describe the *2 COQ benchmarks* which we discussed as being useful in PSP management.
- Describe & discuss the difference in yield management associated with *defect injection vs. removal*
- Name the *5 categories of defect causes* & discuss how the PSP's PIP might relate to them.

## Appendix B: Design Notation

- Be familiar with the various *design notations* discussed in the textbook and in class. Be able to *interpret / explain* designs represented in each type of notation. And be able to *document* designs using each of these notations. These notations include set algebra, boolean algebra, and Karnaugh maps, as well as the 4 design templates discussed in chapters 10 & 12.
- Describe what is meant by a “proper” *state machine*. Describe the three requirements for a state machine to be “proper”: can reach a return (final) state from every other state, all states are complete and orthogonal, & all state transitions are complete & orthogonal.
- Explain the *practical significance* of the three conditions of a “proper” state machine.

## Ch. 10: Software Design

- Define *software design* & discuss the *2 aspects of design quality*
- Identify the *place in the PSP* where design quality is addressed.
- Discuss the *iterative* nature of SW development, & specifically that of SW design. Describe how the *SASY* approach differs from Humphrey’s.
- Describe & diagram the *hierarchical* nature / structure of design.
- Explain what *requirements definitions & design specifications* are & their key differences.
- Discuss why *natural languages* (such as English) are less than ideal for use as design notations, and how *semi-formal* notations can be helpful.
- Name & discuss *design notation criteria*
- Name & describe the *4 design templates* used in the PSP. The State Specification Template actually has both graphical & tabular forms. Describe the graphical version (as well as the tabular one) & explain why both forms are useful.
- Be able to *fill in* the templates from a given scenario, & *draw* state transition diagrams. Be able to *interpret / describe* templates which are given to you.
- Give an *alternate name* by which Operational Scenarios are known.
- Describe / discuss the *relationship* between multiple design levels and an iterative process.

## Ch. 11: Scaling Up the PSP

- Describe what it means to *scale up* a PSP.
- Discuss at least 2 *reasons* why it is necessary to continually be able to scale up your PSP.
- Name & describe the 5 *stages of product size* & the associated process capabilities which are necessary with each.
- Describe / define what it means for a *SW system* to be *scaleable*.
- Discuss *pros & cons* of abstractions / patterns.

## Ch. 12: Design Verification

- Name and explain the 7 *types of design verification methods* discussed by Humphrey. (See slide 5 of chapter 12.)
- Be able to *determine* if a relatively simple state machine (given in either graphical or tabular form) is “*proper*” or not.

## Ch. 13: Defining the SW Process

- Name & describe the *basic SW process elements*
- Discuss the role of *prioritized goals* in process development and assessment.
- Name & describe the 5 *perspectives* on what you do and where you want to be. Explain the differences between these 5 perspectives, and how convergence is related to them. (See slide 18 of chapter 13.)
- Discuss / describe how to *validate* your initial process.
- Name, explain, & describe the various *aspects* which should be *included* when *defining a process phase*
- Explain how *process evolution* takes place.

## Ch. 14: Using the PSP

- Discuss various roles which *commitment* plays in your daily use of the PSP.
- Identify, list, and describe the 3 *potential costs* of using a PSP.
- Name and explain the 4 *stages of learning* and relate these to Bohn’s stages of knowledge.
- Describe how *coaching* might benefit software developers, & the objectives a SW coach might have.