Multiple-Choice Questions:

1. What often-false assumption does the waterfall model made about requirements specifications?

    a. Specifications are predictable

    b. Specifications are stable

    c. Specifications have low change rates

    d. All of the above

    e. None of the above

2. True or false? It is better to discover defects later in the development process. That way, you can have more of the system finished before you worry about fixing things.

    a. True

    b. False

3. An empirical process model iterates between…

    a. … feedback and adaptation

    b. … design and implementation

    c. … requirements gathering and design

    d. … user studies and testing

    e. None of the above

4. True or false? In iterative software development, it is recommended that iterations be 3 to 6 months in length.

    a. True

    b. False

5. If your project has unstable requirements (i.e., that are prone to change), you should use a waterfall process model.

    a. True

    b. False

6. In iterative development, how long should an iteration generally be?

    a. 1 week

    b. 2–6 weeks

    c. 2–4 months

    d. 6 months to a year

    e. None of the above

7. Which of the following is meant by a *software development process*?

    a. A running instance of a program; for example, a UNIX process is a softw. devel. process

    b. Something developers do to accomplish a goal during a project; for example, planning poker is a softw. devel. process for estimation

    c. Something developers use to accomplish a goal during a project; for example, Subversion is a softw. devel. process for configuration management

    d. A structure imposed on the development of a software product; for example, developing iteratively and incorporating best practices might be ingredients in a softw. devel. process

    e. None of the above

8. Which one of the these is a bad length for an iteration?

    a. 1 week
    b. 2 weeks
    c. 4 weeks
    d. 6 weeks
    e. All of the above

9. Which one of these is appropriate in an agile and iterative development process?
    a. Gather a complete set of requirements before designing/building anything.
    b. Implement the backend of the system first—that is, before implementing the frontend functionality with which users interact.
    c. Generate and maintain complete, detailed design documents, which comprehensively model all aspects of the design.
    d. Implement the system incrementally, building it up bit by bit.
    e. Test the code at the end, after the system has been completely implemented.

10. Which of the following is not an agile value?
    a. Individuals and interactions over processes and tools
    b. Working software over comprehensive documentation
    c. Customer collaboration over contract negotiation
    d. Responding to change over following a plan
    e. None of the above (i.e., all are agile values)

11. Which of the following problems does iterative development directly address?
    a. Design erosion
    b. Unstable requirements
    c. Program comprehension
    d. All of the above
    e. None of the above

12. Which type of process control model is appropriate for software development?
    a. A “defined” process control model
    b. An “empirical” process control model
    c. A “remote” process control model
    d. A “parallel” process control model
    e. None of the above

Solutions:

1. d
2. b
3. a
4. b
5. b
6. b
7. d
8. a
9. d
10. e
11. b
12. b

Question:

Answer the following 3 related questions:

- What often-false assumption does the waterfall software engineering process make?
- Why does this false assumption cause considerable problems for waterfall?
- How does iterative development overcome these problems?

Solution:

Waterfall software development makes the false assumption that requirements are mostly stable and can be known from the beginning.

This false assumption creates considerable problems for waterfall because the whole system may be developed before problems with the requirements are discovered. Furthermore, the later defects are discovered in a software product, the more expensive they are to fix (the Defect Cost Increase (DCI) Principle).

Iterative development overcomes these problems by maintaining a tight feedback loop. That is, feedback on the system is collected at regular intervals, revealing any problems early in the process when they are less expensive to correct.

Multiple-Choice Questions:

1. Which of the following activities are not done by the developers?

    a. US creation

    b. US corrections

    c. Set priorities of USs

    d. Add Estimations

    e. None of the above

2. In the agile development process taught in class, the development team estimates each user story and decides the priority for each story.

    a. True

    b. False

3. Which of the following techniques is used for estimating effort?

    a. Role playing

    b. Blueskying

    c. Planning poker

    d. Observation

    e. None of the above

4. T or F? The larger the estimate, the more likely it is to be accurate.

    a. True

    b. False

5. T or F? Planning poker uses the "wisdom of the single biggest expert" to estimate how long it will take to implement user stories.

    a. True

    b. False

Solutions:

1. c
2. b
3. c
4. b
5. b

Problem: Describe the process of iteration planning that we used in this course by writing 7 sentences. Create each sentence by filling in 3 blanks with the following words/phrases. Fill in *all* blanks.

a. Blank #1: developer, customer
b. Blank #2: estimates, selects (for iteration), assigns (to developer), creates, prioritizes
c. Blank #3: tasks, user stories

_____ _____ _____ .

_____ _____ _____ .

_____ _____ _____ .

_____ _____ _____ .

_____ _____ _____ .

_____ _____ _____ .

_____ _____ _____ .

Solution:

developer (or customer)	Creates	user stories.
developer	estimates	user stories.
customer	prioritizes	user stories.
developer	selects	user stories.
developer	creates	tasks.
developer	estimates	tasks.
developer	assigns	tasks.

Problem: All else being equal, which of the following USs most likely has the more accurate estimate?

Title: Animated Buttons

Description: Use jQuery to animate buttons.

Estimate: 2 days

Title: Review Flight

Description: A user will be able to leave a review for a shuttle flight they have been on.

Estimate: 20 days

Solution:

US Animate Buttons

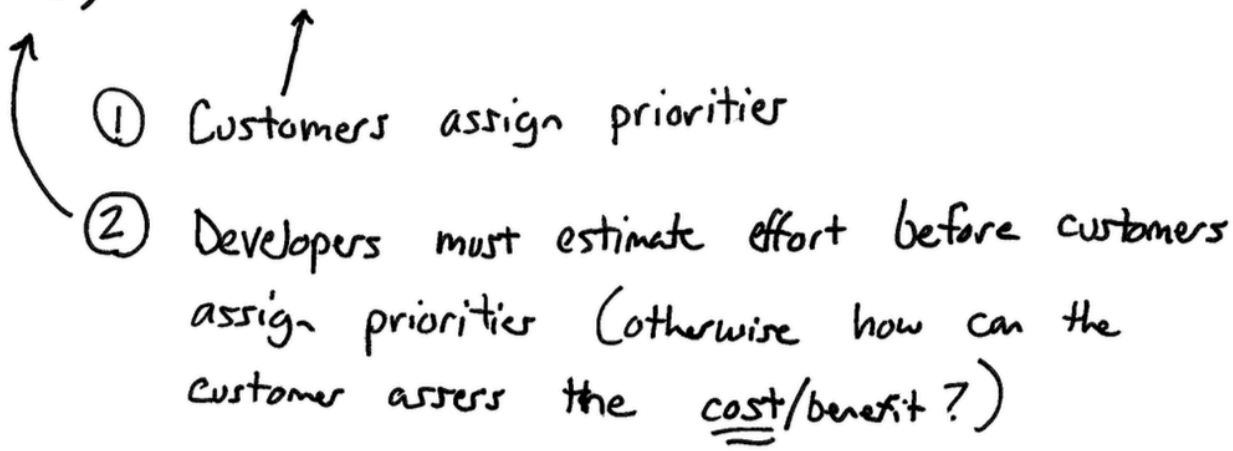
(Because estimates of less than 15 days are generally more accurate than ones over 15 days.)

Problem: What two things are wrong with the following series of steps?

1. First, the developers solicit user stories from the customer.
2. Next, the developers assign a priority level to each user story.
3. Next, the developers estimate the effort required to implement each user story.

Solution:

(1) First, the developers solicit user stories from the customer.
(2) Next, the developers assign a priority level to each user story.
(3) Next, the developers estimate the effort required to implement each user story.



Problem: After your team chooses the USs to implement in an iteration, but before the team begins implementing, what three things must the team do?

Solution:

1. Break the USs into tasks
2. Estimate the time to complete each task
3. Assign each task to a developer