Study and Test Taking Strategies – COMP 311

© 2008 Tim Kington, Todd Whittaker

Studying Strategies

- Remember the pyramid of work in this course design – skipping layers is considered harmful.
Studying Strategies

• Before class:
  • Review previous week’s material.
  • Read the text before the presentation.
  • List basic concepts, theories, methods.
  • Do the examples before reading them.
  • Identify the hard parts in the material.
  • Compose your questions.

• In class:
  • Listen, think, and follow.
  • Participate in classroom discussions.
  • Thinking trumps note taking.
  • Ask questions on the hard parts.
  • Think in advance, if you can.
  • Try to answer others’ questions.
Studying Strategies

• After class:
  • Review the class material and ask, “What have I learned from the class?”
  • Work many exercises and ask, “What have I learned from the exercises?”
  • Find out what you did wrong in the exercises, and re-do them.

• Before a test:
  • Know all the basics.
  • Solve the typical problems.
  • Practice problems under test constraints (time limited, no notes).
  • Try to guess the test questions with your classmates.
Studying Strategies

• After a test:
  • “What did I do wrong, and why?”
  • “What have I learned from the test?”

• After the course:
  • “What have I learned from the course?”
  • “What can I do now that I couldn’t before?”

Test Taking Strategies

• Sleep well the night before.
• Arrive early and relaxed.
• Read through the entire test before answering any questions.
• Budget your time for each question.
  • Determine points per minute
  • Answer fast, high point value first.
Test Taking Strategies

- Always write down something
- Show your work
- Reread the question and your answer to ensure that you have answered all parts of the question
- Anticipate what the test author will ask.
Test Taking Strategies

- Anticipate what the test author will ask.
- A story...

UML Notation

- On pages 751-760 in Koffman
UML Notation

SourceClass
uses
DestinationClass

UML Notation

SourceClass
extends
DestinationClass
SourceClass implements DestinationClass

SourceClass has a DestinationClass
SourceClass has a DestinationClass

Closed diamond is "composition." DestinationClass doesn’t exist separately from SourceClass.

SourceClass has a DestinationClass
SourceClass has a DestinationClass

Open diamond is "aggregation." DestinationClass can exist separately from SourceClass.