

C173/D278 Study Questions by Competency

How to use this document

This document is a great final review tool after you have completed all of the course material. It does not replace the exercises and activities in Zybooks. You will need to be familiar looking at and interpreting pseudocode. It can help you determine if you have a good understanding of the concepts from each lesson.

After completing the preassessment, target the areas below competency and use these questions to help identify what is clear and where there may be missing details in your understanding.

Use the zyBooks material to look up any questions you are unable to answer and if you cannot find the answer in the lesson email us at cmcomputerscience@wgu.edu.

Introduction to Computer Programming

The graduate examines basic computer programming elements, including data types, constants, variables, operators, and expressions.

Lesson 2 Questions | Programming Fundamentals

- What is the purpose of a variable?
- How do you save a value to a variable?
- What is a programming expression?
- What is an identifier?
- What constitutes a valid identifier?
- What is a literal?
- What is an operator?

- What precedence rules does programming use?
- How does an integer differ from a float?
- What happens if you divide two integers? A integer and float?
- What happens if you divide a nonzero floating point number by zero?
- How do you convert an item's type?
- What does the modulo operator do?
- What is the difference in a variable and a constant?
- How does an array work?
- What does index reference?
- What is the purpose of each different data type?

Basic Constructs of Programming

The graduate determines how to achieve programming goals through functions and control structure.

Lesson 3 Questions | Control Structures

- How does a branch differ from a loop?
- How does an if-else branch work?
- What does the equality operator do? Does it work for all data types?
- What are the four relational operators?
- What are the three logical operators?
- Explain the precedence rules.
- How is an infinite loop created?
- What is a sentinel value?
- What are the three parts of a loop?
- What is difference in a while loop and do/while loop?

- Which control structure is guaranteed to run only one time?
- Which loop is used when you are unsure how many times you may need to iterate?
- Which loop is used when you know how many times you want to iterate?

Lesson 4 Questions | Functions

- What is a function?
- Why are functions useful?
- What is a function call?
- What are parameters and arguments?
- What happens if you define a function but do not call it?
- What does a return statement do?

Algorithms

The graduate interprets algorithms.

Lesson 5 Questions | Algorithms

- What is a binary search?
- What is a linear search?

The Design Process

The graduate describes steps of the software design process.

Note: Review the [Software Design & UML Overview](#)

Lesson 6 Questions | Software Design process

- What is SDLC?
- What are the four phases of SDLC?
- What activities takes place in each of the four phases?
- In which phase do you write code?

- How does a waterfall approach differ from an agile approach?

Lesson 7 Questions | Unified Modeling Language

- What is UML?
- Which UML diagrams are structural?
- Which UML diagrams are behavioral?
- Which UML diagrams are activity?
- Which UML diagrams are used in the analysis phase?
- Which UML diagrams are used in the design phase?
- Which UML diagrams are used in the implementation phase?
- Which UML diagrams are used in the testing phase?
- What is a use case diagram used for?
- What is a class diagram used for?
- What is a sequence diagram used for?

Programming Languages

The graduate compares various scripting and programming languages.

Note: Review the [Language Survey Overview](#)

Lesson 8 Questions | Language Survey

- What is the difference in a compiled language and an interpreted language?
- Which languages are compiled?
- Which languages are interpreted?
- What is the difference in statically typed and dynamically typed languages?
- Which languages are statically typed?
- Which languages are dynamically typed?

- What is the difference in an object-oriented language and a non object-oriented language?
- Which languages are object oriented?
- Which languages are not object oriented?
- How does a markup language differ from a programming or scripting language?
- What is a programming library?
- Why are programming libraries used?
- Are libraries compiled or precompiled?