

Computer Programming

General Assessment Information

Blueprint Contents

General Assessment Information	Sample Written Items
Written Assessment Information	Performance Assessment Information
Specific Competencies Covered in the Test	Sample Performance Job

Test Type: The Computer Programming assessment is included in NOCTI's Teacher assessment battery. Teacher assessments measure an individual's technical knowledge and skills in a proctored proficiency examination format. These assessments are used in a large number of states as part of the teacher licensing and/or certification process, assessing competency in all aspects of a particular industry. NOCTI Teacher tests typically offer both a written and performance component that must be administered at a NOCTI-approved Area Test Center. Teacher assessments can be delivered in an online or paper/pencil format.

Revision Team: The assessment content is based on input from subject matter experts representing the following states: California, Connecticut, and Pennsylvania.



11.0201- Computer
Programming/Programmer,
General



Career Cluster 11-
Information Technology



15-1131.00- Computer
Programmers

Written Assessment

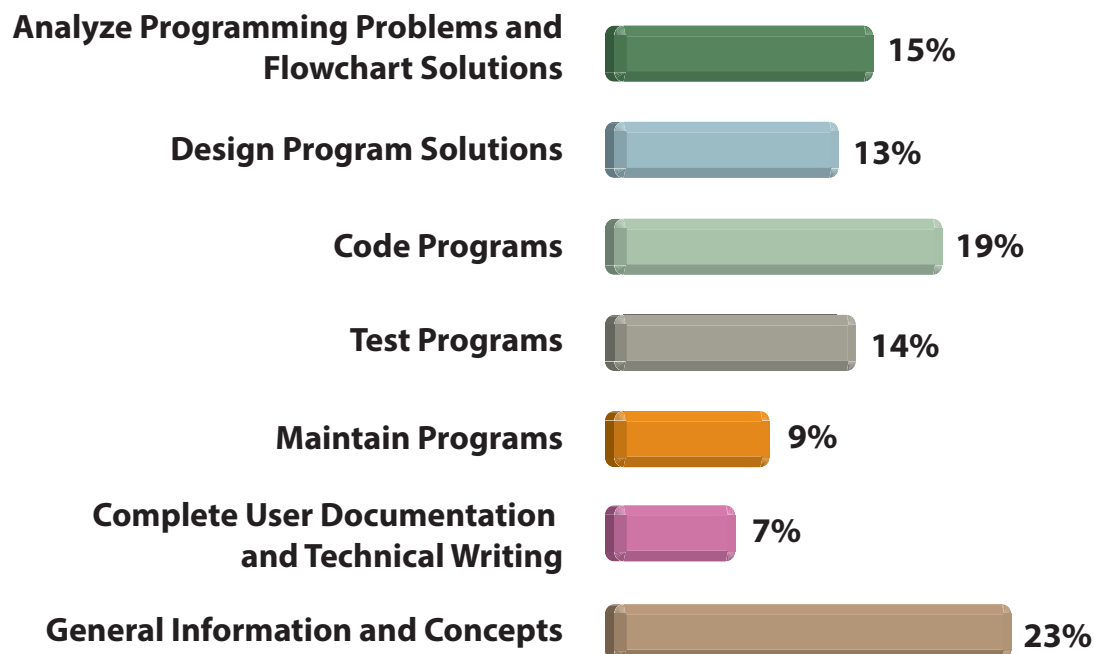
NOCTI written assessments consist of questions to measure an individual's factual theoretical knowledge.

Administration Time: 3 hours

Number of Questions: 160

Number of Sessions: This assessment may be administered in one, two, or three sessions.

Areas Covered



Specific Standards and Competencies Included in this Assessment

Analyze Programming Problems and Flowchart Solutions

- Analyze user requirements for a given outcome
- Determine input and output formats for a program
- Determine the flow of data through network
- Identify and describe a data flow diagram
- Identify and describe a process logic diagram
- Describe the system development cycle (i.e., code management, ongoing revisions)

Design Program Solutions

- Determine where data is to be accessed/stored
- Design data storage and layout
- Apply principles of quality, efficient programming
- Explain the importance of a design review
- Apply implementation plans for a new system
- Assess ongoing impact of existing systems

Code Programs

- Determine the variables and data types for a program
- Prepare and code routines using structured logic
- Identify various programming languages
- Apply appropriate computer language syntax
- Explain unit testing requirements
- Document appropriate comments and programmer notes



(Continued on the following page)

Specific Standards and Competencies (continued)

Test Programs

- Explain system testing requirements
- Design and analyze test plan for use in program testing
- Test programs and evaluate results for accuracy
- Correct programming errors discovered during testing
- Identify appropriate debugging tools

Maintain Programs

- Change existing programs when requirements change
- Correct existing program errors
- Update documentation for existing programs
- Provide user instructions on program modifications



(Continued on the following page)

Specific Standards and Competencies (continued)

Complete User Documentation and Technical Writing

- Develop documentation narrative
- Document data use and storage
- Develop online help for users

General Information and Concepts

- Apply general design and programming concepts
- Identify various hardware platforms and run-time environments
- Identify human aspects in information systems
- Identify general information technology (IT) definitions and terms
- Adhere to best programming practices and methodologies
- Exhibit understanding of data hierarchy, access methods, and manipulation



Sample Questions

Data that is represented in a tagged-format language is

- A. delimited
- B. fixed-length
- C. XML
- D. binary

Large programs used by many different people should be stored on a

- A. server
- B. personal computer
- C. DVD drive
- D. tape backup

Each module in top-down programming should

- A. be well distributed
- B. represent a loop
- C. represent a program function
- D. contain a procedure call

Test data should be developed that will

- A. execute the program properly the first time
- B. validate the operating system
- C. contain only invalid data
- D. generate the answers wanted by users

Documentation standards should be

- A. changed frequently
- B. defined up front
- C. dictated by the end users
- D. determined by the programmer

Performance Assessment

NOCTI performance assessments allow individuals to demonstrate their acquired skills by completing actual jobs using the tools, materials, machines, and equipment related to the technical area.

Administration Time: 3 hours

Number of Jobs: 2

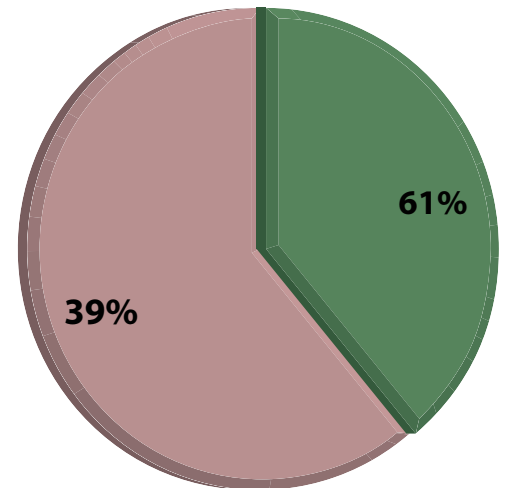
Areas Covered:

61% Write a Program

Create an order form to track CD purchases from a website, set up a GUI panel-type form, enter each set of given test data, perform appropriate calculations, display results in a table on the GUI panel, print out source code and output report.

39% Design Solution Logic

Read the provided programming situation; create a flowchart of pseudocode that solves the programming situation.



Sample Job

Design Solution Logic

Maximum Time: 30 minutes

Participant Activity: The participant will be provided a programming situation. Create a flowchart or pseudocode that solves the programming situation (problem definition).

