

## Competency Framework Examples

### Sample Tasks

Sample tasks have been listed here for various combinations of component, activity and constraint for the three representative competency areas: software, systems and applications. Many, but not all the tasks are atomic. The tasks are not restricted to those that require CS or KA Core topics only.

#### Software Competency Area

Component – Activity	Tasks with constraint(s) <i>italicized</i>
Program – Design	<ul style="list-style-type: none"> <li>● Design <i>efficient</i> data structures for a problem</li> <li>● Design test cases to determine if a program is <i>functionally correct</i></li> <li>● Design an API for a <i>service</i></li> </ul>
Program – Develop	<ul style="list-style-type: none"> <li>● Write a program that <i>meets a given specification</i></li> <li>● <i>Automate</i> testing of new code under development</li> <li>● Develop a program that <i>leverages libraries and APIs</i></li> <li>● <i>Work in a team effectively</i> to solve a problem</li> </ul>
Program – Document	<ul style="list-style-type: none"> <li>● Document a program</li> <li>● <i>Consistently</i> format source code.</li> </ul>
Program – Evaluate	<ul style="list-style-type: none"> <li>● Evaluate an existing application (open source or proprietary) as a whole or partial solution for <i>meeting a defined requirement</i></li> </ul>
Program – Maintain	<ul style="list-style-type: none"> <li>● Refactor a program.</li> <li>● Perform code review to evaluate the quality of code</li> </ul>
Program – Humanize	<ul style="list-style-type: none"> <li>● Ensure <i>fair and equitable access</i> in a program</li> <li>● Document the <i>accountability, responsibility, and liability</i> an individual/company assumes when releasing a given service/software/product</li> <li>● Incorporate <i>legal and ethical privacy requirements</i> into a given service/software/product’s development cycle</li> <li>● Convey the <i>benefits of diverse development teams and user bases</i> on the services/software/products the company provides, as well as the impacts that a lack of diversity can have on them</li> </ul>
Program – Improve	<ul style="list-style-type: none"> <li>● Debug a program</li> </ul>
Program – Research	<ul style="list-style-type: none"> <li>● Compute the running time of a program</li> <li>● Formally prove the <i>correctness</i> of code</li> </ul>
Algorithm – Design	<ul style="list-style-type: none"> <li>● Design an <i>efficient</i> algorithm for a problem</li> </ul>
Algorithm – Document	<ul style="list-style-type: none"> <li>● Explain how an algorithm <i>satisfies a set of requirements</i></li> </ul>

Algorithm – Evaluate	<ul style="list-style-type: none"> <li>Evaluate the <i>efficiency</i> of an algorithm</li> </ul>
Algorithm – Maintain	<ul style="list-style-type: none"> <li>Redesign an algorithm to <i>improve a non-functional requirement</i></li> </ul>
Algorithm – Humanize	<ul style="list-style-type: none"> <li>Justify that an algorithm provides <i>fair and equitable access</i> to data</li> </ul>
Algorithm – Research	<ul style="list-style-type: none"> <li>Prove the <i>correctness</i> of an algorithm</li> <li>Compute the <i>run time efficiency</i> of an algorithm</li> </ul>
Language/Paradigm – Design	<ul style="list-style-type: none"> <li>Select an <i>appropriate</i> language/paradigm for an application</li> </ul>
Language/Paradigm – Document	<ul style="list-style-type: none"> <li>Justify the choice of a language/paradigm for a program</li> <li>Write a white paper to describe how a program is translated into machine code and executed</li> <li>Write a white paper explaining how a program executes in an <i>efficient manner</i> with respect to memory and CPU utilization</li> </ul>
Language/Paradigm – Evaluate	<ul style="list-style-type: none"> <li>Evaluate the <i>appropriateness</i> of a language/paradigm for an application</li> <li>Explain the benefits and challenges of converting an application into <i>parallel/distributed versions</i></li> <li>Write a white paper explaining how a program effectively utilizes language features to make it <i>safe and secure</i></li> </ul>

### Systems Competency Area

Component – Activity	Tasks with constraint(s) <i>italicized</i>
Processor – Design	<ul style="list-style-type: none"> <li>Revise a specification to enable <i>parallel processing</i> without violating other essential properties or features</li> </ul>
Processor – Develop	<ul style="list-style-type: none"> <li>Develop a version of CPU-based application to <i>run on a hardware accelerator</i> (GPU, TPU, NPU)</li> <li>Implement a <i>parallel/distributed</i> version of a known algorithm</li> </ul>
Processor – Evaluate	<ul style="list-style-type: none"> <li>Evaluate the <i>performance-watt</i> of a machine learning model deployed on an <i>embedded device</i></li> </ul>
Processor – Improve	<ul style="list-style-type: none"> <li>Identify and repair a <i>performance</i> problem due to sequential bottlenecks</li> </ul>
Storage – Evaluate	<ul style="list-style-type: none"> <li>Assess the <i>performance implications</i> of cache memories in an application</li> <li>Apply knowledge of operating systems to assess page faults in CPU-GPU memory management and their <i>performance impact</i> on an accelerated application</li> </ul>
I/O – Design	<ul style="list-style-type: none"> <li>Design software modules for <i>sensor hardware integration</i></li> </ul>
I/O – Develop	<ul style="list-style-type: none"> <li>Develop a sensing-actuator robotics arm for an automated manufacturing cell</li> <li>Develop a benchmarking software tool to assess the <i>performance gain in removing I/O bottlenecks</i> in code</li> </ul>

Communication – Design	<ul style="list-style-type: none"> <li>• Design a networking protocol.</li> <li>• Design software that enables <i>safe</i> communication between processes</li> </ul>
Communication – Develop	<ul style="list-style-type: none"> <li>• Develop a networked application</li> <li>• Deploy and securely operate a network of wireless sensors.</li> <li>• Develop software that enables <i>safe communication</i> between processes</li> </ul>
Communication – Evaluate	<ul style="list-style-type: none"> <li>• Evaluate the performance of a network, in specific <i>latency, throughput, congestion, and various service levels</i></li> </ul>
Communication – Maintain	<ul style="list-style-type: none"> <li>• Defend a network from an <i>ongoing distributed denial-of-service</i> attack</li> </ul>
Communication – Humanize	<ul style="list-style-type: none"> <li>• Write a white paper to explain <i>social, ethical, and professional issues</i> governing the design and deployment of a networked system</li> </ul>
Communication – Improve	<ul style="list-style-type: none"> <li>• Identify failures in a datacenter network</li> <li>• Identify and repair a <i>performance problem</i> due to communication or data latency</li> </ul>
Architecture – Develop	<ul style="list-style-type: none"> <li>• Deploy a system in a <i>cloud environment</i></li> <li>• Deploy an application component on a <i>virtualized container</i></li> </ul>
Architecture – Evaluate	<ul style="list-style-type: none"> <li>• Find the <i>performance bottleneck</i> of a given system architecture</li> </ul>
Data – Design	<ul style="list-style-type: none"> <li>• Design how a new application’s data will be stored</li> </ul>
Data – Develop	<ul style="list-style-type: none"> <li>• Create a database for a new application</li> </ul>
Data – Maintain	<ul style="list-style-type: none"> <li>• Get data back online after a disruption (e.g., power outage)</li> </ul>
Data – Humanize	<ul style="list-style-type: none"> <li>• Produce a white paper assessing the <i>social and ethical implications</i> of collecting and storing the data from a new (or existing) application</li> <li>• Assess the <i>legal and ethical implications</i> of collecting and using customer/user data</li> </ul>
Data – Improve	<ul style="list-style-type: none"> <li>• Improve a database application’s <i>performance (speed)</i></li> <li>• Modify a concurrent system to use a more <i>scalable, reliable or available</i> data store</li> </ul>

### Applications Competency Area

Component – Activity	Tasks with constraint(s) <i>italicized</i>
Input – Design	<ul style="list-style-type: none"> <li>• Design an <i>intuitive</i> user interface for an application</li> </ul>
Input – Develop	<ul style="list-style-type: none"> <li>• Implement the user interface of an application</li> </ul>
Input – Humanize	<ul style="list-style-type: none"> <li>• Write a paper on the <i>accessibility</i> of a user interface</li> </ul>
Computation – Design	<ul style="list-style-type: none"> <li>• Specify the operators and partial-order planning graph to solve a logistics problem, showing all ordering constraints</li> </ul>
Computation – Develop	<ul style="list-style-type: none"> <li>• Implement an agent to play a <i>two-player complete information</i> board game</li> </ul>

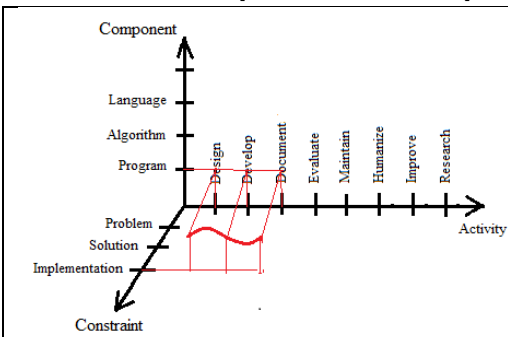
	<ul style="list-style-type: none"> <li>• Write a program that uses Bayes rule to predict the probability of disease given the conditional probability table and a set of observations</li> <li>• Train a neural network to play a video game (e.g., Mario, Atari)</li> </ul>
Computation – Evaluate	<ul style="list-style-type: none"> <li>• Compare the <i>performance</i> of three supervised learning models on a dataset</li> <li>• Explain some of the <i>pitfalls</i> of deep generative models for image or text and how this can affect their use in an application</li> </ul>
Computation – Humanize	<ul style="list-style-type: none"> <li>• Write an essay on the effects of data set bias and how to mitigate them</li> </ul>
Platform – Design	<ul style="list-style-type: none"> <li>• Determine whether to develop an app as a <i>native app</i> or as a <i>cross-platform app</i></li> </ul>
Platform – Develop	<ul style="list-style-type: none"> <li>• Create a mobile app that provides a <i>consistent user experience</i> across various devices, screen sizes, and operating systems</li> <li>• Develop a <i>secure web interface</i> for a business application</li> </ul>
Platform – Evaluate	<ul style="list-style-type: none"> <li>• Evaluate the <i>usability and accessibility</i> of an immersive system</li> </ul>
Platform – Improve	<ul style="list-style-type: none"> <li>• Optimize a <i>dynamic web site</i> for <i>evolving business needs</i></li> </ul>

## Sample Competency Specifications

The following are some sample competency specifications for tasks that require various knowledge areas and skills. They have been listed under the three competency areas, that is, Software, Systems, and Applications.

### Software Competency Area

- **Task Software1:** Develop test cases to determine if a program is functionally correct.
- **Competency statement:** Develop test cases and test a given program.
- **Required knowledge:**
  - SDF-Practices
- **Required skills:** Develop
- **Desirable professional dispositions:** Meticulous, Persistent, Responsible



This is a compound task: the component is a set of test cases for a program; the activities are design, develop, and document; and the constraint is verifying the correctness of implementation of the program.

- **Task Software2:** Perform code review for a teammate.
- **Competency statement:** Communicate clearly and collaboratively to provide feedback to a teammate about a piece of code.
- **Required knowledge:**
  - SDF-Practices
  - SE-Teamwork
  - SE-Validation
- **Required skills:** Apply
- **Desirable professional dispositions:** Collaborative, Communicative, Meticulous

- **Task Software3:** Work on a team effectively.
- **Competency statement:** Focus on long-term team dynamics and communicate effectively.
- **Required knowledge:**
  - SE-Teamwork
- **Required skills:** Apply
- **Desirable professional dispositions:** Collaborative, Communicative, Proactive, Responsive

- **Task Software4:** Make an informed decision regarding which programming language/paradigm to select and use for a specific application.
- **Competency statement:** Apply knowledge of multiple programming paradigms, including their strengths and weaknesses relative to the application to be developed, and select an appropriate paradigm and programming language.
- **Required knowledge:**
  - FPL-OOP
  - FPL-Functional
  - FPL-Logic
  - FPL-Event-Driven
  - FPL-Types
  - FPL-Translation
  - FPL-Pragmatics
  - SPD-Embedded
  - FPL- Constructs
- **Required skills:** Explain, Evaluate
- **Desirable professional dispositions:** Inventive

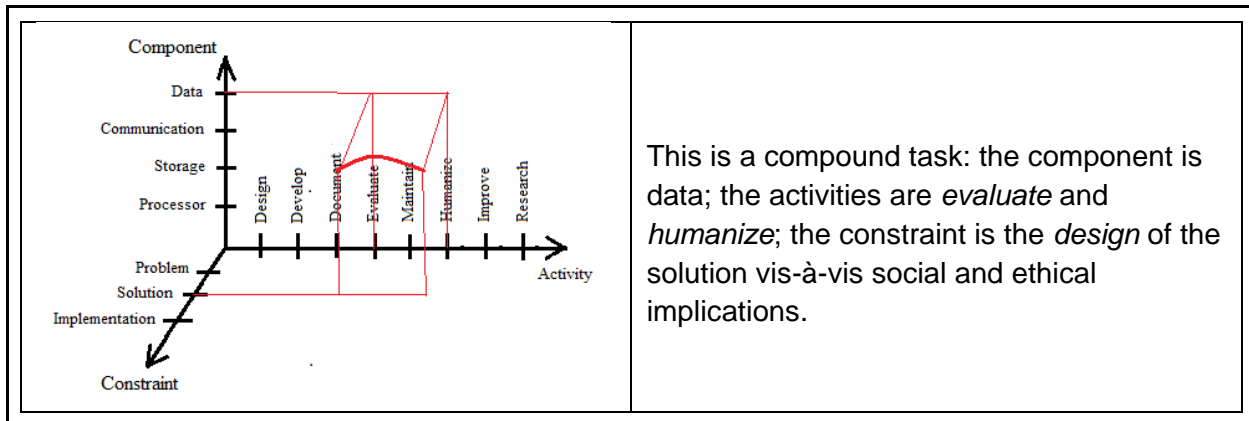
- **Task Software5:** Effectively use a programming language's type system to develop safe and secure software.

- **Competency statement:** Apply knowledge of static type rules for a language to ensure an application is safe, secure, and correct.
- **Required knowledge:**
  - FPL-Types
- **Required skills:** Develop
- **Desirable professional dispositions:** Inventive, Meticulous

### Systems Competency Area

- **Task Systems1:** Develop a version of a CPU-based application to run on a hardware accelerator (GPU, TPU, NPU).
- **Competency statement:** Apply knowledge from systems design to accelerate an application code and evaluate the code speed-up.
- **Required knowledge:**
  - AR-Heterogeneity
  - PDC-Programs
  - SF-Design
- **Required skills:** Evaluate, Develop
- **Desirable professional dispositions:** Meticulous, Inventive

- **Task Systems2:** Produce a white paper assessing the social and ethical implications of collecting and storing data from a new (or existing) application.
- **Competency statement:** Identify the stakeholders and evaluate the potential long-term consequences of the collection and retention of data objects. Consider both potential harm from unintended data use and from data breaches.
- **Required knowledge:**
  - SEP-Context
  - SEP-Ethical-Analysis
  - SEP-Privacy
  - SEP-Professional-Ethics
  - SEP-Security
  - SEP- DEIA
  - DM-Data
  - SEC-Foundations
- **Required skills:** Evaluate, Explain
- **Desirable professional dispositions:** Meticulous, Responsible, Proactive



- **Task Systems3:** Secure data from unauthorized access.
- **Competency statement:** Create database views to ensure data access is appropriately limited.
- **Required knowledge:**
  - DM-Data
  - DM-Relational
  - DM-Processing
  - SEP-Security
  - SEP-Professional-Ethics
  - SEP-Privacy
  - SEC-Foundations
- **Required skills:** Develop
- **Desirable professional dispositions:** Meticulous, Proactive

- **Task Systems4:** Create a database for a new application.
- **Competency statement:** Design the data storage needs (data modeling), assess the social and ethical implications for collecting and storing the data, determine how to store a new application's data (RDBMS vs NoSQL), and create the database, including appropriate indices.
- **Required knowledge:**
  - DM-Data
  - DM-Core
  - DM-Modeling
  - DM-Relational
  - DM-NoSQL
  - DM-Internals
  - SEP-Context
  - SEP-Ethical-Analysis
  - SEP-Privacy
  - SEP-Professional-Ethics

- SEP-Security
- SEP- DEIA
- SEC-Foundations
- **Required skills:** Develop
- **Desirable professional dispositions:** Inventive, Meticulous, Responsible

- **Task Systems5:** Evaluate the performance of a network.
- **Competency statement:** Evaluate the latency, throughput, congestion, and various service levels of a network.
- **Required knowledge:**
  - NC-Applications
  - NC-Routing
- **Required skills:** Evaluate
- **Desirable professional dispositions:** Meticulous, Proactive

- **Task Systems6:** Deploy an application component on an operating system/ runtime/virtualized operating system/container.
- **Competency statement:** Identify and mitigate potential problems with deployment; automate setup of deployment environment; set up monitoring of component execution.
- **Required knowledge:**
  - OS-Purpose
  - OS-Principles
  - OS-Concurrency
  - OS-Scheduling
  - OS-Process
  - OS-Memory
  - OS-Protection
  - AR-Assembly
  - FPL-Scripting
- **Required skills:** Apply
- **Desirable professional dispositions:** Meticulous, Persistent, Proactive

- **Task Systems7:** Improve the performance of a sequential application or component by introducing parallelism.
- **Competency statement:** Evaluate how and when parallelism can improve (or not improve) performance well enough to identify opportunities, as well as implement them and measure the results.
- **Required knowledge:**
  - PDC-Evaluation

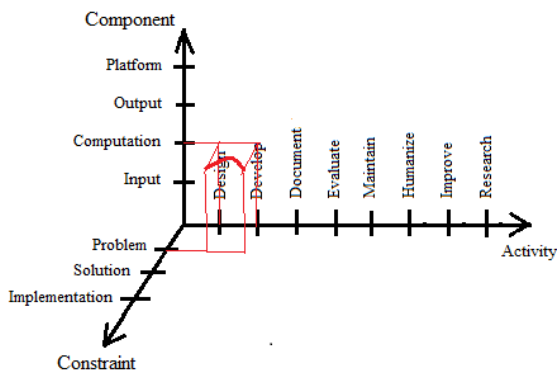


- FPL-Parallel
- **Required skills:** Evaluate, Develop
- **Desirable professional dispositions:** Meticulous, Persistent, Proactive

- **Task Systems8:** Find the performance bottleneck of a given system.
- **Competency statement:** Given a system and its target deployment environment, find its performance bottleneck (e.g., memory, CPU, networking) through analytical derivation or experimental study.
- **Required knowledge:**
  - SF-Performance
  - SF-Evaluation
  - SF-Design
  - SF-Overview
- **Required skills:** Apply, Evaluate, Develop
- **Desirable professional dispositions:** Meticulous, Persistent

### Applications Competency Area

- **Task Applications1:** Implement an agent to make strategic decisions in a two-player adversarial game with uncertain actions (e.g., a board game, strategic stock purchasing).
- **Competency statement:** Use minimax with alpha-beta pruning, and possible chance nodes (expectiminimax), and heuristic move evaluation (at a particular depth) to solve a two-player zero-sum game.
- **Required knowledge:**
  - AI-Search
  - AI-KRR
- **Required skills:** Apply, Develop
- **Desirable professional dispositions:** Inventive, Persistent



This is a compound task: the component is computation, i.e., an algorithm designed for the specific problem of two-player game; the activities are design and develop; the constraints are imposed by the problem – adversarial game with uncertain actions.

- **Task Applications2:** Analyze tabular data (e.g., customer purchases) to identify trends and predict variables of interest.
- **Competency statement:** Use machine learning libraries, data preprocessing, training infrastructures, and evaluation methodologies to create a basic supervised learning pipeline.
- **Required knowledge:**
  - AI-ML
  - AI-SEP
- **Required skills:** Apply, Develop
- **Desirable professional dispositions:** Meticulous, Persistent, Responsible

- **Task Applications3:** Critique a deployed machine learning model in terms of potential bias and correct the issues.
- **Competency statement:** Understand, recognize, and evaluate issues of data set bias in AI, the types of bias, and algorithmic strategies for mitigation.
- **Required knowledge:**
  - AI-ML
  - AI-SEP
- **Required skills:** Explain
- **Desirable professional dispositions:** Responsible

- **Task Applications4:** Visualize a region's temperature record.
- **Competency statement:** Given weather data for a region, design and implement an animation depicting temperature changes over time.
- **Required knowledge:**
  - GIT-Fundamentals
  - GIT-Rendering
  - GIT-Visualization
  - HCI-Design
  - HCI-User
- **Required skills:** Apply, Evaluate, Develop
- **Desirable professional dispositions:** Inventive, Persistent

- **Task Applications5:** Evaluate and provide recommendations to improve a user-facing system.
- **Competency statement:** Apply knowledge of usability, accessibility, and inclusivity to evaluate a user-facing system.
- **Required knowledge:**
  - HCI-User
  - HCI-Accessibility
  - HCI-Evaluation

- HCI-Design
- HCI-SEP
- **Required skills:** Evaluate
- **Desirable professional dispositions:** Meticulous, Responsible

- **Task Applications6:** Determine the aspects of an implementation that require revision to support internationalization.
- **Competency statement:** Evaluate a system to identify culturally-relevant or language-relevant text, symbols, and patterns that may vary by locale.
- **Required knowledge:**
  - HCI-User
  - HCI-Accountability
  - HCI-Accessibility
  - HCI-Evaluation
- **Required skills:** Evaluate
- **Desirable professional dispositions:** Meticulous, Proactive, Responsible

- **Task Applications7:** Document the professional implications of a service/software/product for the company that produced it.
- **Competency statement:** Gather information regarding accountability, responsibility and liability assumed by a company when releasing a service/software/product and present it in a coherent and actionable manner.
- **Required knowledge:**
  - SEP-Professional-Ethics
  - SEP-IP
  - SEP-Privacy
  - SEP-Communication
  - SEP-DEIA
- **Required skills:** Explain
- **Desirable professional dispositions:** Meticulous, Proactive, Responsible

- **Task Applications8:** Determine whether to develop an app as a native app or as a cross-platform app.
- **Competency statement:** Understand performance and scalability issues, and evaluate different approaches and tools by carefully considering factors such as app requirements, target audience, time-to-market, and costs.
- **Required knowledge:**
  - SE-Tools
  - SPD-Common
  - SPD-Mobile
- **Required skills:** Explain

- **Desirable professional dispositions:** Inventive, Meticulous, Responsible

- **Task Applications9:** Build and optimize a secure web page for evolving business needs using a variety of *appropriate* programming languages.
- **Competency statement:** Evaluate potential security hazards and apply optimization techniques.
- **Required knowledge:**
  - AR-Performance-Energy
  - NC-Security
  - OS-Protection
  - SF-Security
  - SE-Design
  - SE-Tools
  - SPD-Common
  - SPD-Mobile
  - SEP-Privacy
- **Required skills:** Develop
- **Desirable professional dispositions:** Adaptable, Meticulous, Proactive, Responsible