Computational science competency

Version: 0.003

Updated: VJ 18th Apr 2025

There are 5 groups:

A. High performance computing

B. Computing

C. Law and compliance

D. Research

E. Soft skills

A. High performance computing

1. Estimate benefit of HPC on specific use case

2. Running specific software/tool/workflow on HPC environments

3. Performance optimization on HPC

4. Debugging software/workflow on HPC

5. Operating HPC system

B. Computing

1. Working remotely in Linux environment via command line interface

2. Managing Python and its packages via package manager

3. Managing scientific software and preparing a related environment

4. Using specific tools, software, or IDEs as each individual or together

5. Estimating computing resource requirement

6. Distribute software, data, or trained model

C. Law and compliance

1. Compliance with license, policy, and ethics

D. Research

1. Methodology, process, and tools

2. Literature Review

3. Data management

4. Scientific data analysis

E. Soft skills

1. Teamwork

2. Communication

3. Human-driven service provision and support

Reference:

https://competency.ebi.ac.uk/framework/bioexcel/3.0/competencies and

https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005772

Red – AI generated content [using local LLM] (will be refined later)

Blue – fast thinking and subjected to be edited later

Black – may has minor change layer

A.1. Estimate benefit of HPC on specific use case

Description

The ability to assess the potential benefits of using HPC to solve a specific problem or improve a process.

Remember

Recalling relevant benefits of HPC on common use cases

Understand

Understanding the basic capabilities and limitations of HPC systems for common use cases.

Apply

Applying knowledge to evaluate the benefit of HPC for specific use cases

Analyze

Identifying trade-offs and considering alternative solutions to determine the best approach for a specific use case.

Evaluate

Assessing the overall impact of using HPC on the problem or process, including its benefits and costs.

Create

Developing new methods or approaches to predict the accurate benefit of HPC in a specific case.

A.2. Running specific software/tool/workflow on HPC environments

Description

The ability to deploy and execute specific software or tools on HPC systems, such as running a simulation, analyzing data, or optimizing an algorithm.

Remember

Deploying and executing specific software or tools on HPC systems by using template script with only minor change in input filename without manually setting an environment.

Understand

Describing the meaning of each line of job submission script and understanding how job schedule works

Apply

Modifying scripts based on given guidelines and configuring the environment to meet specified requirements.

Analyze

Analyzing the implications of changes in job submission scripts and describing the relationship between parameter values and HPC hardware.

Evaluate

Evaluating the quality of job submission scripts in multiple perspectives, including performance, usability, and reproductivity.

Create

Creating the job submission scripts from scratch for deploying and executing unseen software or tools

A.3. Performance optimization on HPC

Description

The ability to optimize the use of HPC resources by improving performance.

Remember

Understand

Apply

Analyze

Identifying potential issues or bottlenecks that may arise during execution and proposing solutions to optimize performance and improve accuracy.

Evaluate

Assessing the overall effectiveness of the optimization process on the HPC system, including any improvements in performance and results.

Create

Developing new methods or approaches to improve the performance of parallel programs on HPC systems.

A.4. Debugging software/workflow on HPC

Description

The ability to debugging software/workflow using HPC resources.

Remember

Recalling the principles and techniques of debugging, including how to identify errors, locate their sources, and correct them.

Understand

Knowing how to apply debugging techniques to specific software or workflow on an HPC system, including preparing input files, configuring parameters, and setting up the environment.

Apply

Running the software or workflow on an HPC system with debugging tools to identify and correct errors.

Analyze

Identifying potential issues or bottlenecks that may arise during execution and proposing solutions to optimize performance and improve accuracy.

Evaluate

Assessing the overall effectiveness of the debugging process on the HPC system, including any improvements in performance and results.

Create

Developing new methods or approaches to optimize the use of HPC resources, including improving performance, reducing costs, and achieving better results.

A.5. Operating HPC system

The ability to operate high-performance computing (HPC) system, such as supercomputers or clusters, for a given purpose.

Knowledge

Comprehension

Application

Analysis

Evaluation

Creation

B.1. Working remotely in Linux environment via command line interface

Remember

- Remember common Linux command line. Being able to understand the basic concepts and commands of a Linux operating system and their usage.

Understand

- Understand how each command works. Knowing how to use common Linux commands to navigate file systems, manage files and directories, and perform other common tasks.

Apply

- Using command to perform tasks. Applying knowledge of Linux commands to solve practical problems and complete tasks in a remote environment.

Analyze

- Understanding the underlying principles and concepts of Linux system administration and troubleshooting and being able to apply this understanding to solve complex problems.

Evaluate

- Being able to critically evaluate the efficiency and effectiveness of different Linux tools and commands in a remote environment, and make informed decisions about which ones to use for specific tasks.

Create

- Developing new strategies and approaches for working remotely in a Linux environment, and sharing knowledge and best practices with others.

Skill

- Linux command line interface

- Linux file system

- Remote connection to a Linux system

Course

- Introduction to Linux for HPC

B.2. Managing programming languages and its packages via package manager

The ability to install, update, and remove programming languages and its packages using a package manager

Remember

- Remember tools and command. Understanding the basics of Python package management and the role of package managers in the Python ecosystem.

Understand

- Understand how each command works. Knowing how to use a package manager to install, update, and remove Python packages, and being able to explain the advantages and disadvantages of different package managers.

Apply

- Using command to perform tasks. Applying knowledge of package management to manage dependencies for a Python project, including installation, configuration, and troubleshooting.

Analyze

- Understanding the underlying principles of Python package management, such as versioning and dependency resolution, and being able to apply this understanding to solve complex package management problems.

Evaluate

- Being able to critically evaluate different package managers in terms of their features, performance, and compatibility with specific project requirements.

Create

- Developing new strategies for managing Python packages, such as creating custom package repositories or developing new package management tools.

Skill

- how to use pip

- how to use conda

Course

- Managing Python packages on HPC [prerequisite: B.1 (70%)]

B.3. Managing scientific software and preparing a related environment

Description

The ability to install, configure, and manage scientific software packages and their dependencies in a Linux environment.

Remember

Understand

Knowing how to use package managers to install, update, and remove scientific software packages and their dependencies, and being able to explain the advantages and disadvantages of different package managers.

Apply

Applying knowledge of scientific software management to manage dependencies for a scientific project, including installation, configuration, and troubleshooting.

Analyze

Understanding the underlying principles of scientific software management, such as versioning and dependency resolution, and being able to apply this understanding to solve complex scientific software management problems.

Evaluate

Being able to critically evaluate different scientific software packages and their dependencies in terms of their features, performance, and compatibility with specific project requirements.

Create

Developing new strategies for managing scientific software packages and their dependencies, such as creating custom package repositories or developing new scientific software management tools.

B.4. Using specific tools, software, or IDEs as each individual or together

Description

The ability to use various tools, software, or integrated development environments (IDEs) to simulate, develop, test, and deploy software applications.

Remember

Understand

Understanding the basics of scientific computing and the role of software packages in this field. Understanding the basics of software development and the role of different tools and software in this field. Knowing how to use a specific tool, software, or IDE to develop, test, and deploy software applications, and being able to explain the advantages and disadvantages of different tools and software.

Apply

Applying knowledge of software development tools and IDEs to develop, test, and deploy software applications, including installation, configuration, and troubleshooting.

Analyze

Understanding the underlying principles of software development, such as versioning and dependency resolution, and being able to apply this understanding to solve complex software development problems.

Evaluate

Being able to critically evaluate different software development tools and their dependencies in terms of their features, performance, and compatibility with specific project requirements.

Create

Developing new strategies for software development tools and their dependencies, such as creating custom package repositories or developing new software development tools.

B.5. Estimating computing resource requirement

Description

The ability to accurately estimate the amount of computational resources required for a given task, such as memory, CPU time, and storage.

Remember

Understand

Understanding the basics of computational resource requirements and being able to explain the differences between various types of resources. Knowing how to use estimation techniques to determine the required amount of computational resources for a specific task.

Apply

Applying estimation techniques to estimate the required amount of computational resources for a given task, including installation, configuration, and troubleshooting.

Analyze

Understanding the underlying principles of computational resource requirements, such as memory allocation and CPU scheduling, and being able to apply this understanding to solve complex computational resource estimation problems.

Evaluate

Being able to critically evaluate different estimation techniques in terms of their features, performance, and compatibility with specific project requirements.

Create

Developing new strategies for estimating computational resource requirements, such as creating custom resource estimation tools or developing new resource management techniques.

B.6. Distribute software, data, or trained model

Description

The ability to accurately distribute software, data, or trained models for a given task, including installation, configuration, and troubleshooting.

Remember

Understanding the basics of software, data, or model distribution and being able to explain the differences between various types of resources.

Understand

Knowing how to use estimation techniques to determine the required amount of computational resources for a specific task.

Apply

Applying estimation techniques to estimate the required amount of computational resources for a given task, including installation, configuration, and troubleshooting.

Analyze

Understanding the underlying principles of software, data, or model distribution, such as memory allocation and CPU scheduling, and being able to apply this understanding to solve complex software, data, or model distribution problems.

Evaluate

Being able to critically evaluate different estimation techniques in terms of their features, performance, and compatibility with specific project requirements.

Create

Developing new strategies for software, data, or model distribution, such as creating custom resource estimation tools or developing new software, data, or model management techniques.

C.1. Compliance with laws, policies, and ethics

Description

The ability to conduct duties in accordance with laws, regulations, organizational policies, and ethical standards.

Remember

Able to recall relevant laws, policies, and ethical standards.

Understand

Able to explain the importance of compliance and ethical behavior in the operation/research duties.

Understand consequences of non-compliance with policies and ethical guidelines.

Apply

Applying compliance and safety protocols and ethical guidelines in a practical setting. Follow rules, regulations, and policies in daily tasks.

Analyze

Identifying areas where policies and ethical guidelines can be improved or strengthened and proposing solutions to address these issues.

Evaluate

Making judgments about the effectiveness of policies or ethical guidelines, and providing feedback on how they can be improved or applied more effectively.

Assess the impact of decisions on ethical and legal standards.

Create

Lead initiatives that foster a culture of integrity and accountability.

D.1. Methodology, process, and tools

Description

Familiarity with scientific methodologies, processes, and tools, and the ability to select the most appropriate approach for a given research project.

Remember

Recalling methodologies, processes, and tools used in scientific research.

Understand

Understanding the advantages, disadvantages, and limitations of different methodologies, and being able to explain how they can be benefit to a particular project.

Apply

Selecting the most appropriate methodologies or tools for solving research questions or problems.

Analyze

Profiling the performance of various methodologies or tools in achieving a research goal and identifying areas where improvements can be made.

Evaluate

Evaluating the effectiveness of various methodologies or tools in achieving a research goal.

Create

Developing new methods or tools that advance current research capabilities or resolve particular scientific problems.

D.2. Literature Review

Description

Demonstrating the ability to conduct a comprehensive review of literature on a given topic and synthesize insights to support the development of one's own research.

Remember

Recalling relevant keywords or literature related to a particular research question or topic.

Understand

Understanding the key findings and contributions of literature.

Apply

Applying literature reviews to inform the development of research questions, hypotheses, or study designs.

Analyze

Analyzing the strengths-weaknesses and limitations of literature.

Identifying the knowledge gap of an existing knowledge base.

Evaluate

Evaluating the overall quality or some parts of literature.

Create

Creating new research approaches based on a comprehensive review of the literature and explaining how they add to existing knowledge.

D.3. Scientific Data management and analysis

Description

Efficiently organizing and managing data to facilitate result analysis, inform research insights, support artificial intelligence development, and enhance decision-making.

Remember

Recalling standard practice use for organizing and managing data.

Recalling relevant data types and sources.

Understand

Explain the role of data organization in supporting AI and research insights.

Apply

Apply appropriate approaches, tools or techniques to manage or analyze datasets for analysis or AI development.

Analyze

Analyzing the performance of tools and techniques used in research or development.

Evaluate

Evaluating the efficiency of tools, techniques, or frameworks and identifying areas of improvement.

Create

Design and implement a data management framework tailored to specific research or AI development context.

E.1. Teamwork

Description

Ability to collaborate effectively with others towards a common goal, sharing knowledge, resources, and responsibilities.

Remember

Recall teamwork principles, roles, and responsibilities within a team.

Understand

Understanding different perspectives and needs within a team and adapting communication and collaboration strategies accordingly.

Apply

Participating actively in team activities, contribute ideas, and work collaboratively on tasks.

Analyze

Analyzing team dynamics. Identifying potential conflicts or inefficiencies and proposing solutions for improvement.

Evaluate

Evaluate the effectiveness of teamwork strategies and make recommendations for optimization.

Create

Ability to develop new strategies, tools, or processes that enhance collaboration and productivity.

E.2. Communication

Description

Communication is the exchange of information, ideas, or emotions through verbal, written, or non-verbal means.

Remember

Recalling basic communication principles, such as encoding, decoding, and feedback.

Understand

Understanding different communication styles.

Apply

Ability to convey ideas effectively through written or verbal means, adjusting communication style to suit the audience and situation. Adapting communication strategies to various contexts and recognizing cultural differences in communication practices.

Analyze

Analyzing messages to identify underlying assumptions, detect subtle cues, and interpret hidden meanings in communication.

Evaluate

Evaluating communication effectiveness and suggesting recommendations for improvement.

Create

Developing new communication strategies or techniques that improve clarity, engagement, and understanding.

E.3. Human-driven service provision and support

Description

Providing assistance or resources to individuals or groups in need, guided by empathy, compassion, and a willingness to help.

Remember

Recalling a practical protocol for supporting individuals or groups in need

Understand

Understanding of the importance of human-driven provision and support in various contexts.

Apply

Providing direct or indirect assistance in a manner that is respectful, caring, and non-judgmental.

Analyze

Assessing individual or group needs, including hidden needs, and considering multifactor such as cultural background, personal history, and situational context.

Evaluate

Evaluating the effectiveness of provision and support strategies, making adjustments or improvements as needed.

Create

Developing new approaches to human-centered services and initiatives that promote a supportive and inclusive society.