Homework 5
Designing a Lattice Visualization Tool Using Classes in Python

Student SB, HD, TS
Professor Jeffrey Carruthers
Release Date Oct 29 2024
Due date Nov 5 2024

Objective

Develop a Python program that models and visualizes crystal lattice structures using object-oriented programming principles. The assignment will focus on using Python classes, and optionally, inheritance, to represent atomic arrangements in simple cubic, body-centered cubic, face-centered cubic, and hexagonal close-packed structures.

Background Information

Each lattice structure has a unique arrangement and density, impacting its physical properties.
Simple Cubic (sc): Basic structure with atoms at each corner of a cube.
Body-Centered Cubic (bcc): Adds an additional atom at the center of the cube.
Face-Centered Cubic (fcc): Places additional atoms at each face of the cube.
Hexagonal Close-Packed (hcp): A non-cubic structure with a hexagonal pattern, known for high packing efficiency.
Include references to materials on atomic structures or crystallography to better understand these formations.

Deliverables

This homework's deliverables includes: Python code (.py or .ipynb)
All files are to be submitted on Blackboard

Grading Rubrics

Class Structure Initialization (20 points): Correct setup of lattice attributes and structure handling. Method Implementation (30 points): Accurate implementation of plot lattice and supercell methods. User Interface and Usability (15 points): Well-defined interface, with clear prompts and error handling. Visualization Accuracy (20 points): Correct representation of lattice structures and supercells in 3D plots. Documentation Comments (15 points): Clear code comments, README file, and, if applicable, report documentation. Autograding Requirements
provide examples of "how to use" the submitted software