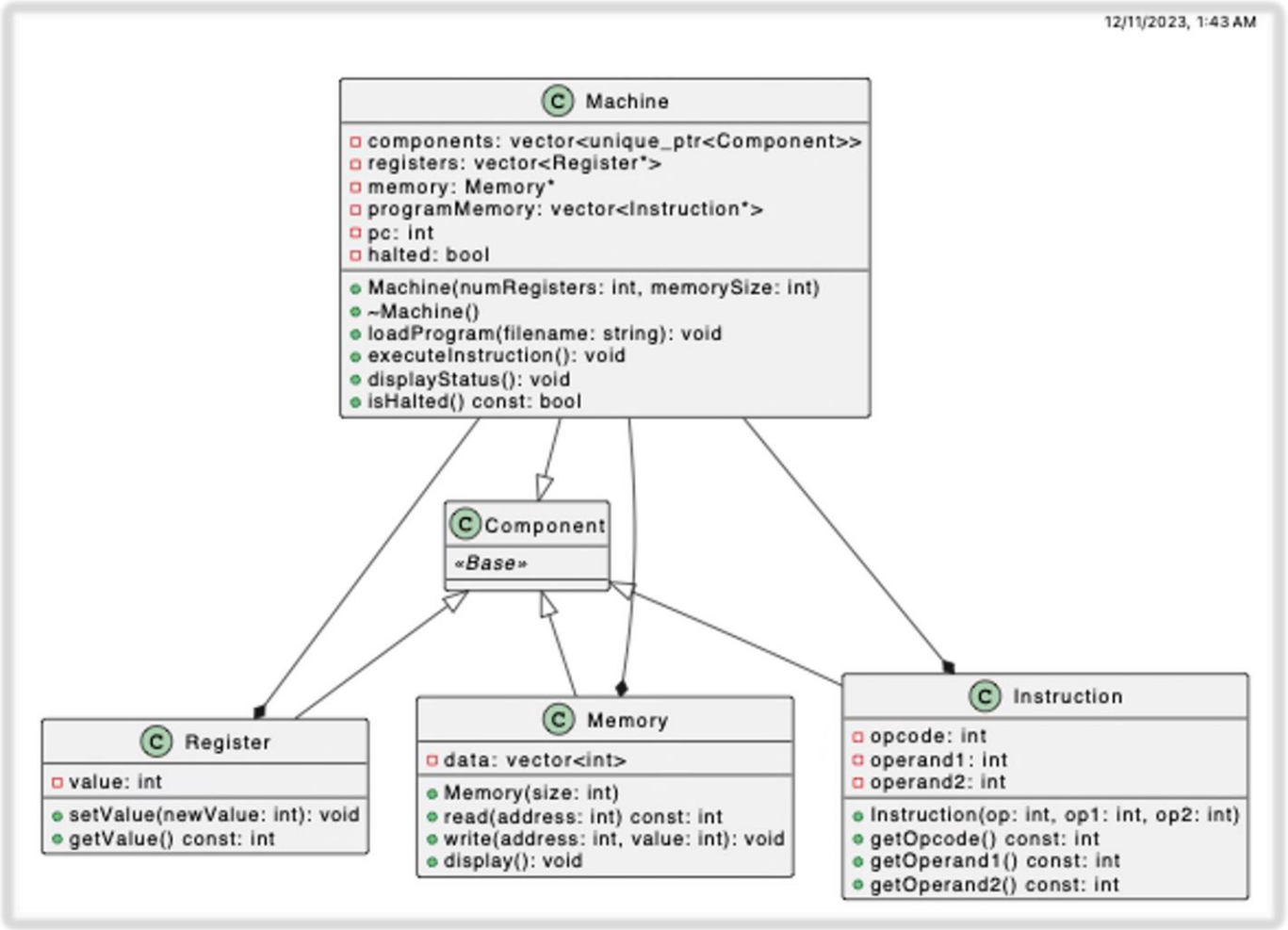
Vole Machine Simulator UML :



Vole Machine Simulator Design description :

1. Class Hierarchy:

- **Component (Abstract Base Class):**

- base class for **Register**, **Memory**, and **Instruction**.

- **Register, Memory, Instruction (Derived Classes):**
  - Each of these classes represents a specific component in the Vole machine emulator. ○ They inherit from the abstract base class **Component**. ○ **Register** represents a register with a value. ○ **Memory** represents memory with read, write, and display operations.
  - **Instruction** represents an instruction with opcode and operands.
- **Machine (Composite Class):**
  - Represents the Vole machine emulator.
  - Has a vector of pointers to **Component** objects (polymorphism). ○ Has specific vectors for **Register**, **Memory**, and **Instruction** objects. ○ Owns the **pc** (program counter) and **halted** state.

2. Encapsulation:

- **Register, Memory, Instruction:**
  - Encapsulate their specific functionalities.
  - Provide methods to set/get values, read/write memory, and access instruction details.
- **Machine:**
  - Encapsulates the entire machine's functionality. ○ Manages the internal state, program counter, and halted state. ○ Controls the execution of instructions and the overall flow of the program.

3. Polymorphism:

- **Machine Class and Vector of Pointers:**
  - Holds a vector of pointers to **Component** objects. ○ This allows storing different types of components (**Register**, **Memory**, **Instruction**) in the same container. ○ Enables polymorphic behavior when accessing and interacting with components.

5. Inheritance:

- **Register, Memory, Instruction Inherit from Component:**
  - Demonstrates the "is-a" relationship, as each of these is a type of **Component**.

6. Composition:

- **Machine Class:**
  - Uses composition to manage instances of **Register**, **Memory**, and **Instruction**. ○ Contains objects of these classes as members.

7. Exception Handling:

- **try-catch Block in main:**
  - Demonstrates handling exceptions for file opening errors or unknown opcodes.