```
"""This module is a simple module to solve linear equations using different
manual/scipy solvers."""

from abc import ABC, abstractmethod
import scipy as sp

from typing import TYPE_CHECKING

if TYPE_CHECKING:
    import numpy as np


class ILinearSolver(ABC):
    @abstractmethod
    def solve(
        self,
        A: "np.ndarray",
        b: "np.ndarray",
        rtol: float,
        max_iter: int,
        M: "np.ndarray",
    ):
        pass


class BiCGStabSolver(ILinearSolver):
    def solve(
        self,
        A: "np.ndarray",
        b: "np.ndarray",
        rtol: float,
        max_iter: int,
        M: "np.ndarray",
    ):
        # ... Use sp.sparse.linalg.bicgstab ...
        x, info = sp.sparse.linalg.bicgstab(A, b, rtol=rtol, maxiter=max_iter,
M=M)
        return x, info


class GMRESSolver(ILinearSolver):
    def solve(
        self,
        A: "np.ndarray",
        b: "np.ndarray",
        rtol: float,
        max_iter: int,
        M: "np.ndarray",
    ):
        x, info = sp.sparse.linalg.gmres(A, b, rtol=rtol, maxiter=max_iter, M=M)
        return x, info
```