

Parallel Leslie Matrix in Age-Structure Model
Module 13.3, “Time after Time—Age- and Stage-Structured Models”

*Introduction to Computational Science:
Modeling and Simulation for the Sciences, 2nd Edition*

Angela B. Shiflet and George W. Shiflet
Wofford College

© 2014 by Princeton University Press

MPI Commands Employed

```
#include <mpi.h>

int MPI_Allreduce ( void *sendbuf, void *recvbuf, int count, MPI_Datatype datatype, MPI_Op op,
    MPI_Comm comm )

int MPI_Barrier( MPI_Comm comm )

int MPI_Bcast ( void *buffer, int count, MPI_Datatype datatype, int root, MPI_Comm comm )

int MPI_Comm_rank ( MPI_Comm comm, int *rank )

int MPI_Comm_size ( MPI_Comm comm, int *size )

int MPI_Finalize()

int MPI_Init(int *argc, char ***argv)

MPI_COMM_WORLD

MPI_DOUBLE

MPI_INT

MPI_IN_PLACE

MPI_SUM
```

General Description
by Matt Beasley

The structure of the serial version of this program must be adjusted for parallelization. The root process does most of the setting up in the program, beginning with parsing the user arguments. This process also loads the values from the input file into the appropriate variables and broadcasts the information to all of the other processes. The majority of the computation done in the program is the matrix multiplication, which is parallelized so that each process calculates a certain number of rows in the resulting matrix. Once all the multiplications are finished, the eigenvalues are calculated, and then, the root process prints the results to the output file.