module Manager

type pair = (int index, double value);
op getTask(result int row, len; result pair [*]elems);
op putResult(int row, len; pair [*]elems);
body Manager

int lengthA[n], lengthC[n];
pair *elementsA[n], *elementsC[n];
# matrix A is assumed to be initialized
int nextRow = 0, tasksDone = 0;

process manager {
    while (nextRow < n or tasksDone < n) {
        # more tasks to do or more results needed
        in getTask(row, len, elems) ->
            row = nextRow;
            len = lengthA[i];
            copy pairs in *elementsA[i] to elems;
            nextRow++;
        [] putResult(row, len, elems) ->
            lengthC[row] = len;
            copy pairs in elems to *elementsC[row];
            tasksDone++;
        ni
    }
}
end Manager

Figure 9.1 (a) Sparse matrix multiplication: Manager process.