

```

proc startIn(int opclass) {
    if (opclass locked)
        insert executing on access list for opclass;
    else
        set the lock;
    dispatcher();
}

proc waitNew(int opclass) {
    look up descriptor of operation class opclass;
    if (new invocations list not empty)
        move new invocations to pending list;
    else {
        insert executing on wait list of opclass;
        if (access list empty)
            clear the lock;
        else
            move first process from access list to ready list;
    }
    dispatcher();
}

proc inDone(int opclass, caller; byte results[*]) {
    if (caller is local)
        call awakenCaller(caller, results);
    else if (caller is remote)
        netWrite(caller's machine, RETURN, (caller,results));
    if (new invocations list of opclass not empty) {
        move invocations from new to pending list;
        move processes (if any) from waiting list to access list;
    }
    if (access list of opclass not empty)
        move first descriptor to ready list;
    else
        clear the lock;
    dispatcher();
}

```

Figure 10.14 Input statement primitives.