```
chan bodies[1:PR](int owner; point p[*], v[*], f[*]);

process Worker[w = 1 to PR] {
  int owner, setSize = n/PR, next = w%PR + 1;
  point p[1:setSize], v[1:setSize], f[1:setSize];
  point tp[1:setSize], tv[1:setSize], tf[1:setSize];
  double m[1:n];
  declarations of other local variables;
  initialize my block of bodies and other variables;
  for [time = start to finish by DT] {
    send bodies[next](w, p[*], v[*], f[*]);
    compute the forces among my block of bodies;
    for [i = 1 to PR-1] {
      receive bodies[w](owner, tp[*], tv[*], tf[*]);
      calculate the forces between my bodies and the new ones;
      send bodies[next](owner, tp[*], tv[*], tf[*]);
    }
    # get back my bodies (owner will equal w)
    receive bodies[w](owner, tp[*], tv[*], tf[*]);
    add forces in tf to those in f;
    update p and v for my set of bodies;
    re-initialize forces on my bodies to zeros;
  }
}
```

Figure 11.14 Pipeline program for the *n*-body problem.