Solutions for CP Exercises, October 22

1. Solution for Andrews Ex. 7.6

```
type Kind = Read | Write;

chan request : (Kind, integer);
chan release : ();
chan readok[i..n] : ();
chan writeok[i..m] : ();

process Reader[i : 1..n] =          process Writer[j : 1..m] =
  ...                                 ...
  send request(Read, i)               send request(Write, j)
  receive readok[i]();                receive writeok[j]();
  reading;                            writing;
  send release();                     send release();
  ...                                 ...


process RWControl =
  var k : kind;
      id : integer;
      active : integer := 0;
  receive request(k, id);
  repeat
    if k = Read then
      while k = Read do
        { send okread[id]();
          active := active + 1;
          receive request(k, id);
        }
    else
      { send okwrite[id]();
        active := 1;
        receive request(k, id);
      }
    while active > 0 do
      {receive release();  active := active − 1}
  forever
```

Rather than accepting all request at any time and record pending request, it has here been chosen to serve requests in order of arrival as long as possible, ie. either serve a single write request or a consecutive sequence of read requests. Releases need only be considered when a new request is about to be served.