task R.SERVEPEERS()
    while True **do**
        Wait for a connection from $P_i^?$
        for $S^t \in S$ **do**
            if $|T^t| < N$ **then** $[S^t] \Rightarrow P_i^?$

task $\mathcal{R}$.UPDATEFULLNESS()
    while True **do**
        $[|\mathcal{T}_j|] \leftarrow \mathcal{S}_j$

task $\mathcal{S}_j$.SERVEPEERS()
    while True **do**
        Wait for connection from $\mathcal{P}_i$
        if $\mathcal{P}_i$ is joining **then**
            for $\mathcal{P}_k \in \mathcal{T}_j$ **do** $[\mathcal{P}_k] \Rightarrow \mathcal{P}_i$
            $\mathcal{T}_j = \mathcal{T}_j \cup \mathcal{P}_i$
        else
            $\mathcal{T}_j = \mathcal{T}_j \setminus \mathcal{P}_i$
        $[|\mathcal{T}_j|] \Rightarrow \mathcal{R}$

task $\mathcal{P}_k$.$(\mathcal{P}_i)$ ▷ Reply a [`hello`] message.
    $[\texttt{hello}] \rightarrow \mathcal{P}_i$

task $\mathcal{P}_i$.JOIN($\mathcal{R}$) ▷ Incoming peer $\mathcal{P}_i$ joins a team.
    $[[\mathcal{S}_j]] \leftarrow \mathcal{R}$.GETSPLITTER($\mathcal{P}_i$) ▷ Receive a splitter.
    for $\mathcal{P}_k \in [[\mathcal{T}_j]] \leftarrow \mathcal{S}_j$.GETPEERS($\mathcal{P}_i$) **do** ▷ Receive a list of peers.
        $[\texttt{hello}] \rightarrow \mathcal{P}_k$
        $[\texttt{hello}] \leftarrow \mathcal{P}_k$ **with timeout**