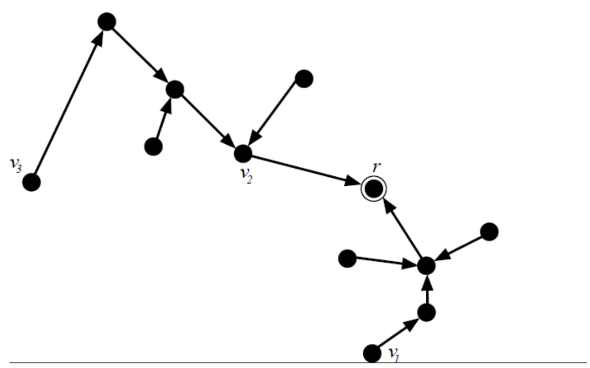
University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn
M. Fuchs, Z. Parsaeian

# Theory of Distributed Systems
# Sample Solution Exercise Sheet 10

## Exercise 1: Concurrent Ivy

Consider the tree for the Ivy shared-variable protocol shown in the following figure. There are three concurrent requests placed by the nodes $v_1$, $v_2$, and $v_3$. The token is initially held by the circled node labeled $r$. We assume synchronous execution.

1. Give the order in which the requests are serviced.
2. Draw the tree after the last request has been served.




## Sample Solution

a) The three nodes are served in the order $v_2, v_3, v_1$.

b) Figure 1 depicts the structure of the tree after the requests have been served. Since $v_1$ is served last, it holds the token at the end.

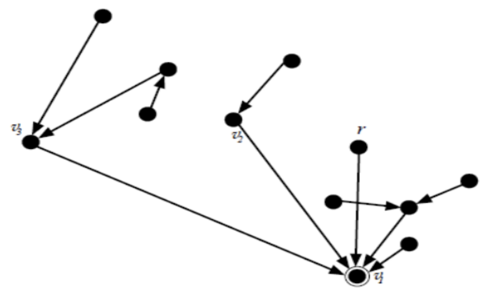



Figure 1: Tree after the requests have been served.

## Exercise 2: Tight Ivy

First, recall the following theorem from the lecture notes: :contentReference[oaicite:1]index=1

**Theorem 1** (sdf)**.** ***If the initial tree is a star, a find request of Ivy shared object algorithm needs at most*** $\log n$ ***steps on average, where*** $n$ ***is the number of processors.***

In this exercise, we show that this bound is tight.

1. Construct a tree of $n$ nodes (with $n$ a power of two) in which each request, when performed sequentially by suitably chosen nodes, indeed requires $\log n$ steps.
2. Show that this worst case can even occur when each node requests the object exactly once.

## Sample Solution

We define a family of trees $(T_i)_{i \geq 0}$ recursively:

- $T_0$ is a single node.
- For $i > 0$, $T_i$ consists of a root together with $i$ subtrees, which are $T_0, T_1, \dots, T_{i-1}$, as its children.

**Number of nodes.** By induction, the number of nodes in $T_i$ is

$$n = 1 + \sum_{j=0}^{i-1} 2^j = 2^i.$$

**Radius.** Again by induction, since the largest-radius child of the root is $T_{i-1}$,

$$R(T_i) = 1 + R(T_{i-1}) = i.$$

**Cost of a request.** Let $C \colon T_i \to T_{i-1}$ be the operation that "cuts off" the subtree $T_{i-1}$ from the root. A request starting at a node at distance $i$ from the root of $T_i$ traverses the roots of $T_1, T_2, \dots, T_i$. Under the Ivy protocol, each of these becomes a child of the request node and loses its largest child subtree. Thus after the request the tree again has subtrees $T_0, T_1, \dots, T_{i-1}$ below the new root, so its shape is unchanged and every such request costs exactly $i$ steps. Since $n = 2^i$, this is

$$i = \log_2 n.$$