Input: Remaining Nodes after Filtering Process in
Output: Node for the service placement out

```
1: //Handle a provisioning request
2: handler(http.Request){
3:     receivedNodes = decode(http.Request);
4:     receivedPod = decodePod(http.Request);
5:     node = selectNode(receivedNodes, receivedPod);
6:     return node
7: }
8: //Return the best candidate Node (recursive)
9: selectNode(receivedNodes, receivedPod){
10:    targetLocation = getLocation(receivedPod);
11:    minBandwidth = getBandwidth(receivedPod);
12:    min = math.MaxFloat64;
13:    copy = receivedNodes;
14:    // find min RTT
15:    for node in range receivedNodes{
16:        rtt = getRTT(node, targetLocation);
17:        min = math.Min(min, rtt);
18:    }
19:    // find best Node based on RTT and minBandwidth
20:    for node in range receivedNodes{
21:        if min == getRTT(node, targetLocation){
22:            if minBandwidth ≤ getAvBandwidth(node){
23:                return node;
24:            }
25:            else
26:                copy = removeNode(copy, node);
27:        }
28:    }
29:    if copy == null
30:        return null, Error("No suitable nodes found!");
31:    else
32:        return selectNode(copy, receivedPod);
33: }
```