

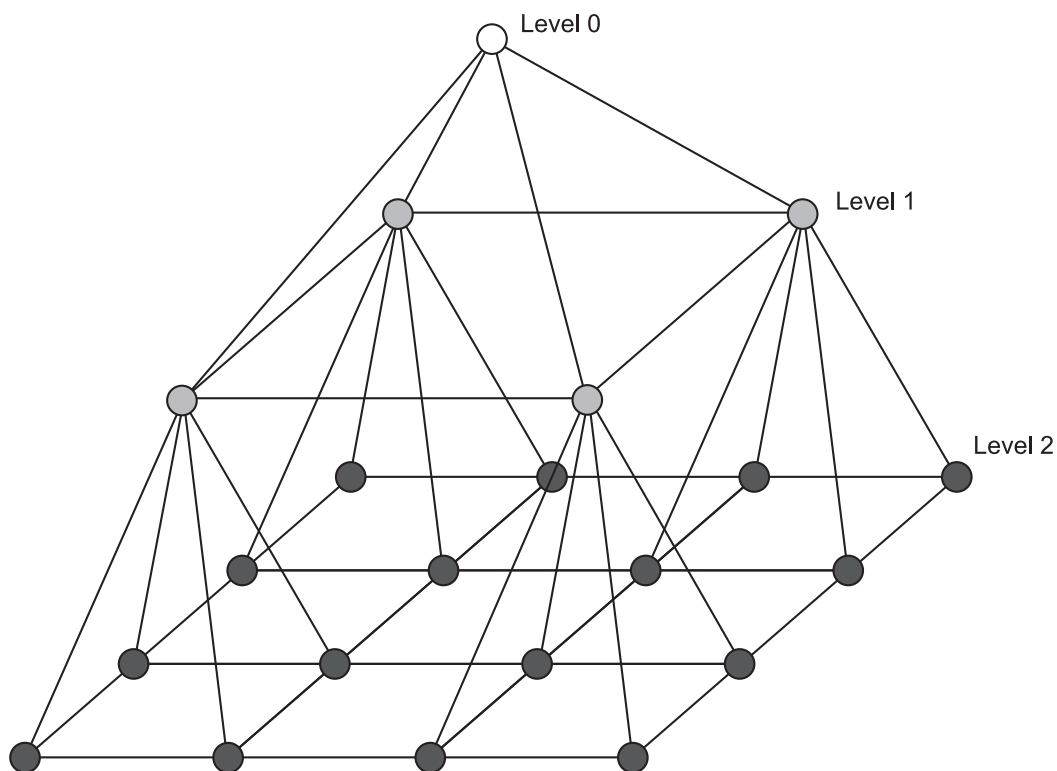
Parallel Programming and HPC

Exercise Sheet 6: Repetition

12th July 2011

1 Network Topologies

The following image shows the first levels of a recursive network topology – a pyramid network. Each node has exactly four child-nodes, all child-nodes are connected through a 2-dimensional grid on the next level.

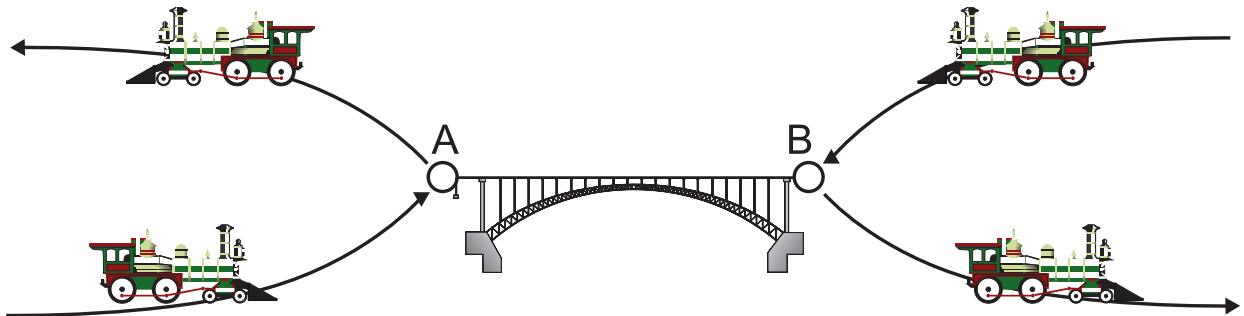


Example of a pyramid-network of height $H = 2$ and a total of $N = 21$ nodes.

- a) Give a general formula, depending only on the height H , to compute the total number of nodes N in the Network. Sums should – if possible – be resolved.
- b) Compute
- I. cost (meaning the number of edges)
 - II. diameter
 - III. bisection width
- of a general pyramid-network depending on N and H only.

2 Semaphores

A railway track between two cities contains a bridge over a canyon that can be accessed by a single train at any point in time only. Hence, this bridge is an exclusively usable resource and needs to be implemented as a critical section.



There exist two types of trains: *R-trains* that always drive from the left-hand to the right-hand side and *L-trains* that always drive from the right-hand to the left-hand side. Give a correct synchronisation (pseudo code) using as many semaphores as necessary, thus, no two trains can access the bridge at the same time and *R-trains* and *L-trains* access the bridge alternately, i.e. after an *R-train* always follows an *L-train* and vice versa. Also give a correct initialisation of all your semaphores!

3 Loop Dependencies

Given is the following code fragment inside two nested loops:

```
for( i=3 ; i <= N ; i++ ) {  
    for( j=1 ; j <= M ; j++ ) {  
S1:        A[i-3][j] = 2 * B[i][j+2];  
S2:        C[i][j]   = D[i, j+1] + A[i][j];  
S3:        B[i][j-1] = C[i][j];  
    }  
}
```

- Examine these statements (according to BERNSTEIN) and name all occurring dependencies! Input dependencies may be neglected.
- State for each dependency named in (a) both distance and direction vector in order to check if it is a *loop-carried* or a *loop-independent* dependency.
- Could the two nested loops be (partially) run in parallel? Justify your answer!

4 OpenMP

On some quadratic meadow the grass needs to be cut. Thus, several people – each equipped with a mowing machine – have to organise themselves in order to do this work in parallel. Consider the meadow as a 2-dimensional array A of size $N \times N$ and some function `mow()` to be executed for each element a_{ij} of A . Write a parallel program using OpenMP and think about sufficient synchronisation!