Homework 6 Solutions

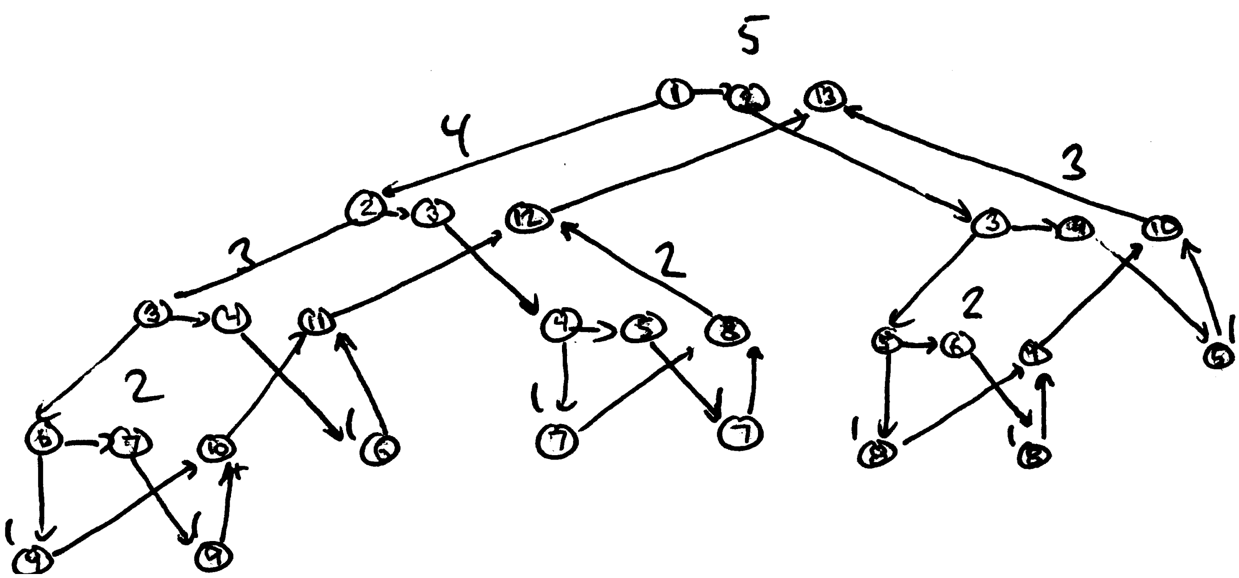
CSC 140 – Advanced Algorithm Design and Analysis

If you find any errors in the solution writeup, please let me know. Ask in class or come to office hours if you need any further help understanding the problems and their solutions.

27.1-1) None. Work (T_1) is unchanged because with one processor all work is done serially and so the spawns have no effect. Span (T_∞) is unchanged because with unlimited processors there is always one available to handle the spawn (the spawning thread is left idle waiting for the sync, but with infinite processors available this is negligible). Since parallelism is defined T_1/T_∞ and neither value is changed, parallelism is unchanged too.

27.1-2) Below is a DAG for P-Fib(5). The work is the number of dots: 29. Just as the span of P-Fib(4) is two longer than the span of P-Fib(3) (ie, the longest path through the call graph is two longer), the span of P-Fib(5) is two longer than the span of P-Fib(4): 10. The parallelism $T_1/T_\infty = 29/10 = 2.9$.

The numbers outside of circles indicate the P-Fib parameter. The numbers inside the circles indicate a greedy scheduling. There are many greedy schedulings. Each time there are more than three strands ready to go, the choice made effects the choices available in the future. This particular scheduling was made by choosing the ready strand that was highest in the tree, with ties going to the leftmost strand. It results in a schedule of length 13 with 7 complete steps and 6 incomplete steps (4 with only one processor scheduled and 2 with two processors scheduled). Note that, as predicted by Equations 27.2, 27.3, and 27.4, $13 \geq 29/3$, $13 \geq 10$, and $13 \leq 17/3 + 10$.



2) *In a divide and conquer algorithm following recurrence relation $T(n) = aT(n/b) + f(n)$ is $T_\infty(n)$ always asymptotically faster than $T_1(n)$? Explain.*

No. In $T_\infty(n)$, the value a is set to 1. So, if $n^{\log_b a}$ dominated $f(n)$ in $T(n)$, then it would no longer do so in $T_\infty(n)$. In such cases $T_\infty(n)$ is asymptotically faster than $T(n) = T_1(n)$. But, if $f(n)$ is the dominant term in $T(n)$, then it will continue to do so in $T_\infty(n)$, making $T_\infty(n)$ and $T(n) = T_1(n)$ asymptotically equal.