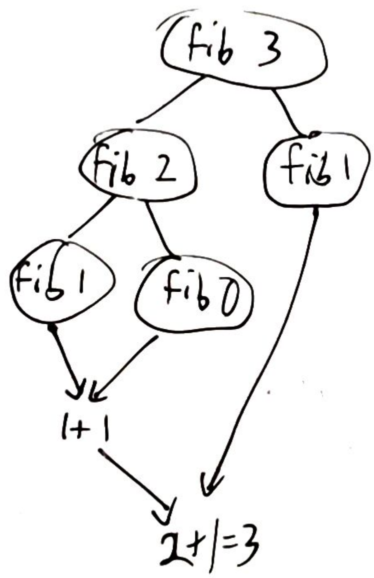
Parallelism

$e ::= \dots \mid \mathrm{par}\ (e_1, e_2)$

$$\frac{e_1 \Downarrow v_1 \qquad e_2 \Downarrow v_2}{\mathrm{par}(e_1, e_2) \Downarrow (v_1, v_2)}$$

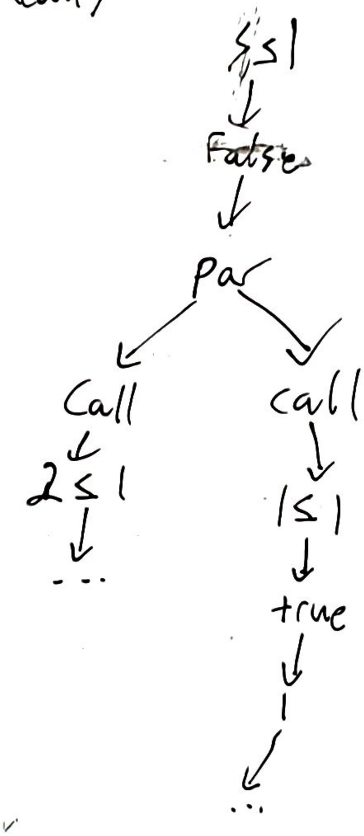
```
fib n =
  if n ≤ 1 then 1
  else
    let (a,b) = par(fib (n-2),
                    fib(n-1))
    in a+b
```

Directed Acyclic Graphs



Really:

1 node = 1 step



Work : $O(\mathrm{fib}(n)) \approx O(\phi^n)$
Span : $O(n)$

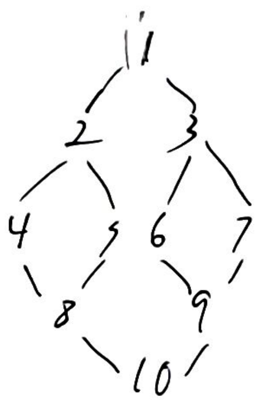
Work: Total amt of computation
# of nodes in DAG
Time to run on 1 proc

Span: "Critical path"
Length of longest path in DAG
Time to run on unlimited procs

schedule - Assign nodes to procs and time steps, respecting deps.
Ideally, want shortest schedule - NP-Hard

	Proc 0	Proc 1
1	fib 3	
2	fib 2	fib 1
3	fib 1	fib 0
4	1+1	
5	2+1	

			Proc 2
1	1		
2	2	3	
3	4	5	6
4	8	7	
5	9		
6	10		



<u>Brent's Theorem</u>: For a DAG w/ work W, span S, on P procs, exists a schedule of length $\le W/P + S\frac{P-1}{P}$

Pf: Take each level at a time.
Time to execute level $i = \left\lceil \frac{\#\text{ of nodes at level } i}{P} \right\rceil$

$$\text{Time} = \sum_{i=0}^{\#\text{levels}} \left\lceil \frac{\#\text{ nodes at } i}{P} \right\rceil \le \sum \left(\frac{\#\text{ nodes at } i}{P} + \frac{P-1}{P} \right)$$

$$\le \frac{\overbrace{\Sigma_i \#\text{ nodes at } i}^{=\text{work}}}{P} + \overbrace{\#\text{ levels}}^{=\text{span}}\left(\frac{P-1}{P}\right)$$

$$= \frac{W}{P} + S\frac{P-1}{P} \quad \text{D}$$

w/in a factor of 2 of optimal

Optimal: $\max(\frac{w}{p}, s)$

Want to know work + span $\Rightarrow$ cost semantics

$e \Downarrow^{(w,s)} v$

$$\frac{}{v \Downarrow^{(0,0)} v} \qquad \frac{e_1 \Downarrow^{(w_1,s_1)} \lambda x.e \quad e_2 \Downarrow^{(w_2,s_2)} v \quad [v/x]e \Downarrow^{(w_3,s_3)} v'}{e_1\ e_2 \Downarrow^{(w_1+w_2+w_3+1,\ s_1+s_2+s_3+1)} v'}$$

$$\frac{e_1 \Downarrow^{(w_1,s_1)} v_1 \quad e_2 \Downarrow^{(w_2,s_2)} v_2}{(e_1,e_2) \Downarrow^{(w_1+w_2,\ s_1+s_2)} (v_1,v_2)} \qquad \frac{e_1 \Downarrow^{(w_1,s_1)} v_1 \quad e_2 \Downarrow^{(w_2,s_2)} v_2}{par(e_1,e_2) \Downarrow^{(w_1+w_2,\ \max(s_1,s_2))} (v_1,v_2)}$$

Want to know cost semantics is correct

$\Rightarrow$ Compare to small-step "bounded implementation" (Blelloch + Greiner) / "Provably efficient implementation" (Harper, Blelloch)

Need a parallel small-step semantics

Interleaving

$$\frac{e_1 \mapsto e_1'}{par(e_1,e_2) \mapsto par(e_1',e_2)} \qquad \frac{e_2 \mapsto e_2'}{par(e_1,e_2) \mapsto par(e_1,e_2')} \qquad \frac{e_1\ val \quad e_2\ val}{par(e_1,e_2) \mapsto (e_1,e_2)}$$

Only counts work

Parallel

$$\frac{e_1 \overset{0,1}{\mapsto} e_1' \quad e_2 \overset{0,1}{\mapsto} e_2'}{par(e_1,e_2) \mapsto par(e_1',e_2')}$$ — Only counts span

Explicit threads

$\overbrace{\sigma}^{\text{threads}}; e \mapsto \sigma'; e'$

"fresh" = not used before

$$\frac{a \text{ fresh} \qquad b \text{ fresh}}{\sigma; \mathrm{par}(e_1, e_2) \mapsto \sigma, a \hookrightarrow e_1, b \hookrightarrow e_2; \mathrm{wait}(a,b)}$$

$$\frac{\sigma(a) = v_1 \quad v_1 \text{ val} \qquad \sigma(b) = v_2 \quad v_2 \text{ val}}{\sigma; \mathrm{wait}(a,b) \mapsto \sigma; (v_1, v_2)}$$

$$\frac{\sigma_0; e_i \mapsto e_i'; \sigma_i' \qquad 1 \le P}{\underbrace{\sigma, a_1 \hookrightarrow e_1, \ldots, a_n \hookrightarrow e_n}_{\sigma_0} \Rightarrow \sigma, \sigma_{1,+}, \sigma_n', a_1 \hookrightarrow e_1', \ldots, a_n \hookrightarrow e_n'}$$

can specify more about schedule