Final Project

• Work on a project of your own choosing (you have about 4 weeks; 3 weeks until presentation).
• Work in a pair of two.
• A 20-25 minutes presentation (during the last two lectures).
• A writeup < 5 pages (single writeup, but in the writeup, specify how responsibilities are divided).
Possible Topics

• You can choose anything that relates to your research (but be sure the deliverable is small enough that it can be done in 4 weeks).
• Implement solution to a problem in Cilk Plus.
• Implement some concurrent data structures and evaluate their performance.
• Literature search on a specific topic that interests you.
• Build a different tool (cache miss? instruction count?)
Project 3: the Cilkprof Algorithm

x: current cursor
u: the cursor pointing to the child where the span goes through
   (but note that span is being updated as the tool executes)
p: span from the first instruction in F to u (green).
c: span along continuation from u to current cursor, x (yellow).
l: the span through a spawned child of F (blue).
Project 3: the Cilkprof Algorithm

x: current cursor
u: the cursor pointing to the child where the span goes through (but note that span is being updated as the tool executes)
p: span from the first instruction in F to u (green).
c: span along continuation from u to current cursor, x (yellow).
l: the span through a spawned child of F (blue).
Project 3: the Cilkprof Algorithm

Upon returning from first spawn.

x: current cursor
u: the cursor pointing to the child where the span goes through
   (but note that span is being updated as the tool executes)
p: span from the first instruction in F to u (green).
c: span along continuation from u to current cursor, x (yellow).
l: the span through a spawned child of F (blue).
Project 3: the Cilkprof Algorithm

\[ \begin{align*}
  x & : \text{current cursor} \\
  u & : \text{the cursor pointing to the child where the span goes through} \\
  & \quad \text{(but note that span is being updated as the tool executes)} \\
  p & : \text{span from the first instruction in F to u (green).} \\
  c & : \text{span along continuation from u to current cursor, x (yellow).} \\
  l & : \text{the span through a spawned child of F (blue).}
\end{align*} \]
Project 3: the Cilkprof Algorithm

x: current cursor
u: the cursor pointing to the child where the span goes through
   (but note that span is being updated as the tool executes)
p: span from the first instruction in F to u (green).
c: span along continuation from u to current cursor, x (yellow).
l: the span through a spawned child of F (blue).

Upon reaching 2\textsuperscript{nd} spawn statement.
Project 3: the Cilkprof Algorithm

p: span from the first instruction in F to u (green).

u: the cursor pointing to the child where the span goes through (but note that span is being updated as the tool executes)

x: current cursor

c: span along continuation from u to current cursor, x (yellow).

l: the span through a spawned child of F (blue).

l2: span through this child.

When returning from 2nd spawn, if (l2+c) > l, update p, u, and l.
Project 3: the Cilkprof Algorithm

x: current cursor
u: the cursor pointing to the child where the span goes through
   (but note that span is being updated as the tool executes)
p: span from the first instruction in F to u (green).
c: span along continuation from u to current cursor, x (yellow).
l: the span through a spawned child of F (blue).