Red-Black Tree Delete

Do normal BST delete; if deleted node has 2 non-leaf children
then moved node takes deleted's color
  do 1 non-leaf delete where moved node was

\[
\begin{array}{c}
\text{if} \\
\end{array}
\]

while \( x \) is doubly black

\[
\begin{array}{c}
\text{if} \\
\end{array}
\]

\[
\begin{array}{c}
\text{if} \\
\end{array}
\]

\[
\begin{array}{c}
\text{if} \\
\end{array}
\]

else if

\[
\begin{array}{c}
\text{else if} \\
\end{array}
\]

\[
\begin{array}{c}
\text{else if} \\
\end{array}
\]

\[
\begin{array}{c}
\text{else if} \\
\end{array}
\]
else if

if furthest sibling red

Done
Graph: represents things and relationships between them
people relationships

nodes/vertices edges

path: seq of verts s.t. adj verts have edge in graph
J6, CH, FM, FDR, JS, FDR, WC

simple path: path w/o repeats J6, CH, FM, FDR, JS

cycle: cycle starting/ending at same place J6, K6, KK, MB, J6, K6, 56, J6

simple cycle: cycle w/o repeat except start, end J6, K6, 56, J6
int foo(int n, int c)
{
    if (n == c)
    {
        return 0;
    }
    int i = 1;
    while (i < n)
    {
        if (i % c == 3)
        {
            if (n % 2 == 1)
            {
                return 0;
            }
        }
        i++; // Code highlighted
    }
}

Vertices: lines of code
Edges: control flow

While (x > 1) do
    x--

is there a path entry-exit that doesn't go through a return?
verts: teams  edge $u \rightarrow v$: $u$ lost to $v$ at least once

what ordering of verbs minimizes edges in "wrong" direction

vertices
edges

FEEDBACK-ARC-SET
(NP-complete)
weighted graph
edges have associated numeric weight
given start/end, what path has lowest total weight
Graph Representation

Adjacency Matrix

<table>
<thead>
<tr>
<th></th>
<th>Y</th>
<th>Col</th>
<th>D</th>
<th>Pr</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>Col</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>D</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>Pr</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>H</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

Adjacency List

Y : 
Col : Y D 
D : Y
Pr : Y D
H : Y col D