Chaitin has proved that register allocation is equivalent to graph coloring and hence NP-complete. Astonishingly, Bouchez et al., Brisk et al., and Hack et al. have proved independently that the interference graph of a program in static single assignment (SSA) form is chordal. A greedy algorithm can optimally color a chordal graph in time linear in the number of edges. Completing the picture, Hack et al. have presented a new SSA-elimination algorithm that does not demand extra registers. The combined result is a simple, optimal, polynomial-time algorithm for a core register allocation problem. The new SSA-elimination algorithm of Hack et al. is essential because classical SSA elimination leads to an NP-complete core register allocation problem, as shown by Palsberg and Pereira. Register allocation with spilling for SSA form remains NP-complete, We can get most of the advantages of SSA form without actually transforming to SSA form: a vast majority of realistic benchmark programs appear to have chordal interference graphs. For example, 95the Java 1.5 library have chordal interference graphs when compiled with the JoeQ compiler. Pereira and Palsberg have shown how to add simple heuristics for spilling and coalescing to the greedy coloring algorithm. The result is a simple and efficient register allocation algorithm which compares well with the iterated register coalescing algorithm of George and Appel.