My program is implemented according to the following sudo code:

resolution(list of input clauses)

candidates = PriorityQueue() # ResPairs, keep sorted on sum of lengths of clauses

for each pair 0<=i<j<len(clause)

if clauses i and j can be resolved

candidates.insert(ResPair(i,j))

while candidates in not empty:

ResPair(i,j) = candidates.pop() # dequeue best candidate

for each proposition p that occurs as opposite literals in clauses i and j:

resolvent = resolve(clauses[i],clauses[j],p)

if resolvent is empty clause: return "success!"

if resolvent is not already in the list of clauses: # if visited, discard

add resolvent to clauses # suppose it gets index m

for each k<m in clauses:

if m and k are resolvable:

candidates.insert(ResPair(k,m))

return "failure"