Quaife's Theorem (O21)

Johan G. F. Belinfante and Claudia D. Huang
2005 July 10

```
In[1]:= SetDirectory["i:"]; << goedel71.09b; << tools.m

        :Package Title: goedel71.09b          2005 July 9 at 11:50 a.m.

        It is now:  2005 Jul 10 at 15:0

        Loading Simplification Rules

        TOOLS.M                       Revised 2005 June 17

        weightlimit = 40
```

summary

The first clause of Quaife's Theorem **(O21)** is already recognized by the **GOEDEL** program. New rewrite rules are derived here to take care of the second clause.

derivation

Lemma:

```
In[2]:= SubstTest[implies, equal[y, nat[z]], or[equal[y, succ[x]],
          member[succ[x], y], not[member[x, y]], not[member[y, omega]]], z → y]

Out[2]= or[equal[y, succ[x]], member[succ[x], y],
          not[member[x, y]], not[member[y, omega]]] == True

In[3]:= or[equal[y_, succ[x_]], member[succ[x_], y_],
          not[member[x_, y_]], not[member[y_, omega]]] := True
```

The following justifies a new conditional rewrite rule:

```
In[4]:= implies[member[y, omega],
           equiv[or[equal[y, succ[x]], member[succ[x], y]], member[x, y]]] // not // not

Out[4]= True

In[5]:= or[equal[y_, succ[x_]], member[succ[x_], y_]] := member[x, y] /; member[y, omega]
```

To facilitate computation, the **GOEDEL** program currently rewrites **2 x** as **x + x**. On account of this, the following rule is needed:

```
In[6]:= SubstTest[member, natmul[nat[x], nat[z]],
         natmul[nat[y], nat[z]], z → succ[set[0]]]

Out[6]= member[natadd[nat[x], nat[x]], natadd[nat[y], nat[y]]] == member[nat[x], nat[y]]

In[7]:= member[natadd[nat[x_], nat[x_]], natadd[nat[y_], nat[y_]]] :=
         member[nat[x], nat[y]]
```

The second clause of **(O21)** is now recognized:

```
In[8]:= member[natadd[set[0], natmul[succ[set[0]], nat[x]]],
         natadd[set[0], natmul[succ[set[0]], nat[y]]]]

Out[8]= member[nat[x], nat[y]]
```