gcd and lcm

Johan G. F. Belinfante
2009 April 2

```
In[1]:= SetDirectory["l:"]; << goedel.09apr01a; << tools.m

        :Package Title: goedel.09apr01a            2009 April 1 at 2:20 p.m.

        It is now:  2009 Apr 2 at 10:20

        Loading Simplification Rules

        TOOLS.M                                    Revised 2009 February 18

        weightlimit = 40
```

summary

In this notebook it is shown that the product of the gcd and lcm of a pair of natural numbers is the product of those numbers.

temporary abbreviations

```
In[2]:= divides[x_, y_] := member[pair[x, y], DIV]

In[3]:= gcd[x_, y_] := APPLY[GLB[DIV], set[x, y]]

In[4]:= lcm[x_, y_] := APPLY[LUB[DIV], set[x, y]]
```

gcd and lcm

Lemma. The gcd divides the lcm.

```
In[5]:= SubstTest[implies, and[divides[u, v], divides[v, w]], divides[u, w],
          {u -> gcd[nat[x], nat[y]], v → nat[x], w -> lcm[nat[x], nat[y]]}] // Reverse

Out[5]= member[pair[APPLY[GLB[DIV], set[nat[x], nat[y]]],
           APPLY[LUB[DIV], set[nat[x], nat[y]]]], DIV] == True

In[6]:= member[pair[APPLY[GLB[DIV], set[nat[x_], nat[y_]]],
           APPLY[LUB[DIV], set[nat[x_], nat[y_]]]], DIV] := True
```

If **x** is not **0**, then **x/gcd(x,y)** and **y/gcd(x,y)** are relatively prime divisors of **lcm(x,y)/gcd(x,y)**, and therefore their product is also a divisor of **lcm(x,y)/gcd(x,y)**.

Lemma.

```
In[7]:= SubstTest[implies,
          and[divides[u, w], divides[v, w], implies[p, equal[set[0], gcd[u, v]]]],
          implies[p, divides[natmul[u, v], w]], {p → not[empty[nat[x]]],
           u -> natdiv[nat[x], gcd[nat[x], nat[y]]], v -> natdiv[nat[y], gcd[nat[x], nat[y]]],
           w -> natdiv[lcm[nat[x], nat[y]], gcd[nat[x], nat[y]]]}] // MapNotNot // Reverse

Out[7]= or[equal[0, nat[x]], equal[0, nat[y]],
          member[pair[natmul[nat[x], nat[y]], natmul[APPLY[GLB[DIV], set[nat[x], nat[y]]],
            APPLY[LUB[DIV], set[nat[x], nat[y]]]]], DIV]] == True

In[8]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Two redundant literals can be removed.

Corollary.

```
In[9]:= SubstTest[and, implies[p, q], or[p, q], {p -> or[equal[0, nat[x]], equal[0, nat[y]]],
          q -> member[pair[natmul[nat[x], nat[y]], natmul[APPLY[GLB[DIV], set[nat[x], nat[y]]],
              APPLY[LUB[DIV], set[nat[x], nat[y]]]]], DIV]}] // MapNotNot

Out[9]= member[pair[natmul[nat[x], nat[y]], natmul[APPLY[GLB[DIV], set[nat[x], nat[y]]],
           APPLY[LUB[DIV], set[nat[x], nat[y]]]]], DIV] == True

In[10]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

The next step is to use the fact that since **(x y)/gcd(x, y) = x (y/gcd(x,y)) = (x/gcd(x,y)) y** is a common multiple of **x** and **y**, it is a multiple of **lcm(x,y)**.

Lemma. Divisibility in the opposite direction.

```
In[11]:= SubstTest[implies,
           and[divides[nat[x], z], divides[nat[y], z]], divides[lcm[nat[x], nat[y]], z],
           z → natdiv[natmul[nat[x], nat[y]], gcd[nat[x], nat[y]]]] // Reverse

Out[11]= member[pair[natmul[APPLY[GLB[DIV], set[nat[x], nat[y]]],
            APPLY[LUB[DIV], set[nat[x], nat[y]]]], natmul[nat[x], nat[y]]], DIV] == True

In[12]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

The final step uses the fact that two natural numbers are equal if each divides the other.

Theorem. The product of the gcd and lcm of two natural numbers is equal to their product.

```
In[13]:= SubstTest[and, divides[u, v], divides[v, u],
          {u -> natmul[APPLY[GLB[DIV], set[nat[x], nat[y]]],
             APPLY[LUB[DIV], set[nat[x], nat[y]]]], v -> natmul[nat[x], nat[y]]}]

Out[13]= equal[natmul[APPLY[GLB[DIV], set[nat[x], nat[y]]],
            APPLY[LUB[DIV], set[nat[x], nat[y]]]], natmul[nat[x], nat[y]]] == True

In[14]:= natmul[APPLY[GLB[DIV], set[nat[x_], nat[y_]]],
           APPLY[LUB[DIV], set[nat[x_], nat[y_]]]] := natmul[nat[x], nat[y]]
```

The **nat** wrappers can be removed.

Corollary.

```
In[15]:= SubstTest[implies, and[equal[x, nat[u]], equal[y, nat[v]]],
           equal[natmul[gcd[x, y], lcm[x, y]], natmul[x, y]], {u → x, v → y}] // Reverse

Out[15]= or[equal[natmul[x, y], natmul[APPLY[GLB[DIV], set[x, y]], APPLY[LUB[DIV], set[x, y]]]],
           not[member[x, omega]], not[member[y, omega]]] == True

In[16]:= or[equal[natmul[x_, y_],
            natmul[APPLY[GLB[DIV], set[x_, y_]], APPLY[LUB[DIV], set[x_, y_]]]],
           not[member[x_, omega]], not[member[y_, omega]]] := True
```