SMALLER for wellorderable sets

Johan G. F. Belinfante
2007 May 3

```
In[1]:= SetDirectory["l:"]; << goedel93.03a; << tools.m

        :Package Title: goedel93.03a      2007 May 3 at 2:55 a.m.

        It is now:  2007 May 3 at 3:2

        Loading Simplification Rules

        TOOLS.M                                    Revised 2007 May 2

        weightlimit = 40
```

summary

A set **x** is **wellorderable** if it is equipollent to some ordinal number. For each such set **x**, the smallest ordinal equipollent to **x** is its cardinality **card[x]**. If **x** is not wellorderable, then **card[x] = V**. The function **CARD** assigns to each wellorderable set its cardinality. The domain of this function is the class **image[Q,OMEGA]** of wellorderable sets.

```
In[2]:= domain[CARD]

Out[2]= image[Q, OMEGA]
```

A wrapper **wob[x]** for wellorderable sets has been introduced. Wellorderable sets are equipollent if they have the same cardinality:

```
In[3]:= member[pair[wob[x], wob[y]], Q]

Out[3]= equal[card[wob[x]], card[wob[y]]]
```

A set **x** is **smaller** than a set **y** if **x** is equipollent to a subset of **y**, but not to **y** itself. The corresponding relation is:

```
In[4]:= class[pair[x, y],
          and[not[member[pair[x, y], Q]], exists[z, and[subclass[z, y], member[pair[x, z], Q]]]]]

Out[4]= SMALLER
```

In this notebook it is shown that a wellorderable set is smaller than another if and only if it has a lesser cardinality.

a membership rule

```
In[5]:= member[pair[x, y], composite[inverse[CARD], z]] // AssertTest

Out[5]= member[pair[x, y], composite[inverse[CARD], z]] ==
         and[member[x, V], member[y, image[Q, OMEGA]], member[pair[x, card[y]], z]]

In[6]:= member[pair[x_, y_], composite[inverse[CARD], z_]] :=
         and[member[x, V], member[y, image[Q, OMEGA]], member[pair[x, card[y]], z]]
```

facts about SMALLER

Lemma. (Simpification rule.)

```
In[7]:= equiv[and[member[x, V], member[pair[x, y], SMALLER]], member[pair[x, y], SMALLER]]

Out[7]= True

In[8]:= and[member[x_, V], member[pair[x_, y_], SMALLER]] := member[pair[x, y], SMALLER]
```

Theorem. A set cannot be smaller than any of its subsets.

```
In[9]:= SubstTest[implies, and[member[u, v], subclass[v, w]], member[u, w],
           {u → pair[x, y], v → SMALLER, w → complement[inverse[S]]}] // Reverse // MapNotNot

Out[9]= or[not[member[pair[x, y], SMALLER]], not[subclass[y, x]]] == True

In[10]:= or[not[member[pair[x_, y_], SMALLER]], not[subclass[y_, x_]]] := True
```

Corollary. No set can be smaller than itself.

```
In[11]:= Map[not,
          SubstTest[implies, member[pair[x, y], SMALLER], not[subclass[y, x]], y → x] // Reverse]

Out[11]= member[pair[x, x], SMALLER] == False

In[12]:= member[pair[x_, x_], SMALLER] := False
```

smaller sets

Theorem.

```
In[14]:= SubstTest[implies, and[member[u, v], subclass[v, w]], member[u, w],
           {u → pair[x, y], v -> composite[inverse[CARD], E, CARD], w -> SMALLER}] // Reverse

Out[14]= or[member[pair[x, y], SMALLER],
           not[member[y, image[Q, OMEGA]]], not[member[card[x], card[y]]]] == True
```

```
In[15]:= or[member[pair[x_, y_], SMALLER],
           not[member[y_, image[Q, OMEGA]]], not[member[card[x_], card[y_]]]] := True
```

Corollary. (Replacing the wellorderable literal with a **wob** wrapper.)

```
In[16]:= SubstTest[or, member[pair[x, t], SMALLER], not[member[t, image[Q, OMEGA]]],
           not[member[card[x], card[t]]], t → wob[y]] // Reverse

Out[16]= or[member[pair[x, wob[y]], SMALLER], not[member[card[x], card[wob[y]]]]] == True

In[17]:= or[member[pair[x_, wob[y_]], SMALLER], not[member[card[x_], card[wob[y_]]]]] := True
```

Theorem.

```
In[18]:= Map[not,
          SubstTest[member, pair[wob[x], wob[y]], intersection[u, v], {u → Q, v → SMALLER}]]

Out[18]= or[not[equal[card[wob[x]], card[wob[y]]]],
           not[member[pair[wob[x], wob[y]], SMALLER]]] == True

In[19]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Lemma.

```
In[20]:= Map[not, SubstTest[member, pair[wob[x], wob[y]], intersection[u, v],
           {u → SMALLER, v -> composite[inverse[CARD], inverse[S], CARD]}]]

Out[20]= or[member[card[wob[x]], card[wob[y]]],
           not[member[pair[wob[x], wob[y]], SMALLER]]] == True

In[21]:= (% /. {x → x_, y → y_}) /. Equal → SetDelayed
```

Main Theorem.

```
In[22]:= equiv[member[pair[wob[x], wob[y]], SMALLER], member[card[wob[x]], card[wob[y]]]]

Out[22]= True

In[23]:= member[pair[wob[x_], wob[y_]], SMALLER] := member[card[wob[x]], card[wob[y]]]
```

a variable-free formulation

Eliminating the **wob** wrappers yields:

```
In[24]:= SubstTest[implies, and[equal[x, wob[u]], equal[y, wob[v]], member[pair[x, y], SMALLER]],
           member[card[x], card[y]], {u → x, v → y}] // Reverse

Out[24]= or[member[card[x], card[y]], not[member[x, image[Q, OMEGA]]],
           not[member[y, image[Q, OMEGA]]], not[member[pair[x, y], SMALLER]]] == True

In[25]:= or[member[card[x_], card[y_]], not[member[pair[x_, y_], SMALLER]],
           not[member[x_, image[Q, OMEGA]]], not[member[y_, image[Q, OMEGA]]]] := True
```

Lemma.

```
In[26]:= SubstTest[empty, dif[u, v],
          {u -> composite[id[image[Q, OMEGA]], SMALLER, id[image[Q, OMEGA]]],
           v -> composite[inverse[CARD], E, CARD]}]

Out[26]= subclass[composite[id[image[Q, OMEGA]], SMALLER, id[image[Q, OMEGA]]],
           composite[inverse[CARD], E, CARD]] == True

In[27]:= % /. Equal → SetDelayed
```

Theorem. A formula for the restriction of **SMALLER** to wellorderable sets.

```
In[28]:= SubstTest[and, subclass[u, v], subclass[v, u],
          {u -> composite[id[image[Q, OMEGA]], SMALLER, id[image[Q, OMEGA]]],
           v -> composite[inverse[CARD], E, CARD]}]

Out[28]= equal[composite[id[image[Q, OMEGA]], SMALLER, id[image[Q, OMEGA]]],
           composite[inverse[CARD], E, CARD]] == True

In[29]:= composite[id[image[Q, OMEGA]], SMALLER, id[image[Q, OMEGA]]] :=
          composite[inverse[CARD], E, CARD]
```