initial segments of omega and OMEGA

Johan G. F. Belinfante
2008 March 9

```
In[1]:= SetDirectory["l:"]; << goedel.08mar08a; << tools.m

        :Package Title: goedel.08mar08a                  2008 March 8 at 3:05 p.m.

        It is now:  2008 Mar 9 at 21:7

        Loading Simplification Rules

        TOOLS.M                                    Revised 2008 February 12

        weightlimit = 40
```

summary

The class of **initial segments** of a well-order relation **w** is defined to be

intersection[complement[set[fix[w]]], invar[inverse[w]], P[fix[w]]].

Reference:

```
In[2]:=  "Karel Hrbacek and Thomas Jech, Introduction to Set Theory,
         Marcel Dekker, Inc., New York, third ed., 1999.  See page 104.";
```

The restriction of the subset relation **S** to the class **omega** of natural numbers is a well-ordering. The class of initial segments for this well-ordering is equal to **omega**. An especially simple derivation of a rewrite rule expressing this fact is derived in this notebook. A similar derivation also works for the case of the class **OMEGA** of all ordinals.

initial segments of omega

Lemma.

```
In[3]:= SubstTest[implies, subclass[u, v],
          subclass[invar[v], invar[u]], {u → composite[inverse[E], id[omega]],
           v → composite[id[omega], inverse[S], id[omega]]}] // Reverse

Out[3]= subclass[invar[composite[id[omega], inverse[S], id[omega]]],
          invar[composite[inverse[E], id[omega]]]] == True

In[4]:= % /. Equal → SetDelayed
```

Lemma.

```
In[5]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
          {t → id[P[omega]], u -> invar[composite[id[omega], inverse[S], id[omega]]],
           v -> invar[composite[inverse[E], id[omega]]]}] // Reverse

Out[5]= subclass[
          intersection[invar[composite[id[omega], inverse[S]]], P[omega]], succ[omega]] == True

In[6]:= % /. Equal → SetDelayed
```

Theorem.

```
In[7]:= equal[succ[omega],
          intersection[invar[composite[id[omega], inverse[S]]], P[omega]]] // AssertTest

Out[7]= equal[intersection[invar[composite[id[omega], inverse[S]]], P[omega]], succ[omega]] ==
         True

In[8]:= intersection[invar[composite[id[omega], inverse[S]]], P[omega]] := succ[omega]
```

Corollary.

```
In[9]:= SubstTest[intersection, invar[t], subvar[t], t -> composite[id[omega], inverse[S]]]

Out[9]= fix[composite[IMAGE[id[omega]], IMAGE[inverse[S]]]] == succ[omega]

In[10]:= fix[composite[IMAGE[id[omega]], IMAGE[inverse[S]]]] := succ[omega]
```

---

initial segments of OMEGA

```
In[11]:= SubstTest[implies, subclass[u, v],
           subclass[invar[v], invar[u]], {u → composite[inverse[E], id[OMEGA]],
            v → composite[id[OMEGA], inverse[S], id[OMEGA]]}] // Reverse

Out[11]= subclass[invar[composite[id[OMEGA], inverse[S], id[OMEGA]]],
           invar[composite[inverse[E], id[OMEGA]]]] == True

In[12]:= % /. Equal → SetDelayed
```

Lemma.

```
In[13]:= SubstTest[implies, subclass[u, v], subclass[image[t, u], image[t, v]],
           {t → id[P[OMEGA]], u -> invar[composite[id[OMEGA], inverse[S], id[OMEGA]]],
            v -> invar[composite[inverse[E], id[OMEGA]]]}] // Reverse

Out[13]= subclass[intersection[invar[composite[id[OMEGA], inverse[S]]], P[OMEGA]], OMEGA] == True

In[14]:= % /. Equal → SetDelayed
```

Lemma.

```
In[15]:= Map[empty[composite[Id, complement[#]]] &, SubstTest[reify, x,
           union[complement[set[ord[x]]], invar[composite[id[w], inverse[S]]]], w → OMEGA]]

Out[15]= subclass[OMEGA, invar[composite[id[OMEGA], inverse[S]]]] == True

In[16]:= subclass[OMEGA, invar[composite[id[OMEGA], inverse[S]]]] := True
```

Theorem.

```
In[17]:= equal[intersection[invar[composite[id[OMEGA], inverse[S]]], P[OMEGA]], OMEGA] //
          AssertTest

Out[17]= equal[OMEGA, intersection[invar[composite[id[OMEGA], inverse[S]]], P[OMEGA]]] == True

In[18]:= intersection[invar[composite[id[OMEGA], inverse[S]]], P[OMEGA]] := OMEGA
```

Corollary.

```
In[19]:= SubstTest[intersection, invar[t], subvar[t], t -> composite[id[OMEGA], inverse[S]]]

Out[19]= fix[composite[IMAGE[id[OMEGA]], IMAGE[inverse[S]]]] == OMEGA

In[20]:= fix[composite[IMAGE[id[OMEGA]], IMAGE[inverse[S]]]] := OMEGA
```