

subclass[P[x],CONST]

Johan G. F. Belinfante
2007 May 11

```
In[1]:= SetDirectory["1:"]; << goedel93.10a; << tools.m

:Package Title: goedel93.10a      2007 May 10 at 8:35 p.m.

It is now: 2007 May 11 at 3:50

Loading Simplification Rules

TOOLS.M                          Revised 2007 May 6

weightlimit = 40
```

summary

A rewrite rule for **subclass[P[x], CONST]** is derived. The key idea is that the class of constant functions can be written as a class of cliques.

derivation

Theorem.

```
In[2]:= SubstTest[cliques, cross[x, y], {x -> V, y -> Id}] // Reverse
```

```
Out[2]= cliques[composite[inverse[SECOND], SECOND]] == CONST
```

```
In[3]:= cliques[composite[inverse[SECOND], SECOND]] := CONST
```

Lemma.

```
In[4]:= subclass[cart[x, x], composite[inverse[SECOND], SECOND]] // AssertTest
```

```
Out[4]= subclass[cart[x, x], composite[inverse[SECOND], SECOND]] ==
  or[and[member[range[x], range[SINGLETON]], subclass[x, cart[V, V]], equal[0, x]]
```

```
In[5]:= subclass[cart[x_, x_], composite[inverse[SECOND], SECOND]] :=
  or[and[member[range[x], range[SINGLETON]], subclass[x, cart[V, V]], equal[0, x]]
```

Theorem.

```
In[6]:= SubstTest[subclass, P[x], cliques[y], y -> composite[inverse[SECOND], SECOND]] // Reverse
```

```
Out[6]= subclass[P[x], CONST] ==  
  or[and[member[range[x], range[SINGLETON]], subclass[x, cart[V, V]], equal[0, x]]]
```

```
In[7]:= subclass[P[x_], CONST] :=  
  or[and[member[range[x], range[SINGLETON]], subclass[x, cart[V, V]], equal[0, x]]]
```