

\$SPAD/src/input aseg7.as

The Axiom Team

July 29, 2014

**Abstract**

# Contents

|   |         |   |
|---|---------|---|
| 1 | License | 3 |
|---|---------|---|

# 1 License

```
-- Copyright (c) 1991-2002, The Numerical ALgorithms Group Ltd.
-- All rights reserved.
--
-- Redistribution and use in source and binary forms, with or without
-- modification, are permitted provided that the following conditions are
-- met:
--
--   - Redistributions of source code must retain the above copyright
--     notice, this list of conditions and the following disclaimer.
--
--   - Redistributions in binary form must reproduce the above copyright
--     notice, this list of conditions and the following disclaimer in
--     the documentation and/or other materials provided with the
--     distribution.
--
--   - Neither the name of The Numerical ALgorithms Group Ltd. nor the
--     names of its contributors may be used to endorse or promote products
--     derived from this software without specific prior written permission.
--
-- THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
-- IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
-- TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
-- PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER
-- OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
-- EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
-- PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
-- PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
-- LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
-- NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
-- SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

\_\_\_ \* \_\_\_

```
#include "axiom.as"
```

```
L ==> List;
```

```
Test2(R:EuclideanDomain) : with {
  constPtrs? :() -> R;
  set : R->R;
  resetConstPtrs :()->R;
} == add {
```

```
  import from R;
  local constantPtrs:R:=0;
```

```

constPtrs?():R == {
  free constantPtrs;
  constantPtrs;
}

set(c:R):R == {
  free constantPtrs;
  constantPtrs := c;
}

resetConstPtrs():R == {
  free constantPtrs;
  constantPtrs := 0;
}
}

Test3 : with {
  constPtrs? : ()-> Integer;
  set: Integer->Integer;
  resetConstPtrs: () ->Integer;

} == add {
  import from Integer;
  constantPtrs:Integer:=0;
  constPtrs?():Integer == {
    free constantPtrs;
    constantPtrs;
  }

  set(c:Integer):Integer == {
    free constantPtrs;
    constantPtrs := c;
  }

  resetConstPtrs():Integer == {
    free constantPtrs;
    constantPtrs := 0;
  }
}

foo2 : with {constPtrs? : ()-> Integer; set: Integer->Integer; resetConstPtrs: () ->Integer } == Test

```

---

## References

- [1] nothing