

\$SPAD/src/input matops.as

The Axiom Team

July 31, 2014

Abstract

Contents

1	Input tests	3
2	License	3

1 Input tests

— input —

```
)compile matops.as
)show MatrixOpSym
matrix [[3,4],[5,6]]
symmetricPart %
```

2 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"
#pile

MatrixOpSym(R:Field) : MOD == Definition where
  Mat ==> Matrix R

  MOD ==> with

    symmetricPart : Mat -> Mat
      ++ \spad{symmetricPart(m)} returns the symmetric part of m

  Definition ==> add

    symmetricPart(m:Mat) : Mat ==
      m1:=transpose m
      import from Integer
      import from R
      inv(2::R) *(m+m1)

```

References

- [1] nothing