FUPRA

0.2

Generated by Doxygen 1.8.14

Contents

Chapter 1

Modules Index

1.1 Modules List

Here is a list of all modules with brief descriptions:

Chapter 2

Data Type Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Data Type Index

3.1 Data Types List

Here are the data types with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

Module Documentation

5.1 abstract_container_array_mod Module Reference

Module that defines an unlimited polymorphic container class and related methods. A container is a fundamental entity allowing to build data structures such as lists and arrays. This is an abstract type, so a derived type must be defined for any specific contents that may be required. Those derived types should provide type-specific methods that require type-guards, such as printing.

Data Types

- type container_array

Functions/Subroutines

- class(∗) function, pointer getvalue (this, index)

  Method that returns returns the requested entry (pointer)
- subroutine putvalue (this, index, value)

  Method that stores a value on the requested index.
- integer function getlength (this)

  Method that returns the length of the array.
- subroutine resizearray (this, newsize, initvalue)

  Method that grows (adds empty space) or shrinks (discards the last entries) of the array. Use sparsely as this might get expensive for large array operations. Should think of a way to use move_alloc()
- subroutine initarray (this, entries, source, initvalue)

  Method that allocates the container array. Deallocates if already allocated.

Variables

- logical, target mc = .true.

5.1.1 Detailed Description

Module that defines an unlimited polymorphic container class and related methods. A container is a fundamental entity allowing to build data structures such as lists and arrays. This is an abstract type, so a derived type must be defined for any specific contents that may be required. Those derived types should provide type-specific methods that require type-guards, such as printing.

Author

Ricardo Birjukovs Canelas

5.1.2 Function/Subroutine Documentation

5.1.2.1 getlength()

```
integer function abstract_container_array_mod::getlength (
            class(container_array), intent(in) this )  [private]
```

Method that returns the length of the array.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this*	

Definition at line 102 of file abstract_container_array.f90.

5.1.2.2 getvalue()

```
class(*) function, pointer abstract_container_array_mod::getvalue (
            class(container_array), intent(in) this,
            integer, intent(in) index )  [private]
```

Method that returns returns the requested entry (pointer)

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this,index*	

Definition at line 68 of file abstract_container_array.f90.

5.1.2.3 initarray()

```
subroutine abstract_container_array_mod::initarray (
             class(container_array), intent(inout) this,
             integer, intent(in) entries,
             type(container), dimension(:), intent(in), optional source,
             class(*), intent(in), optional, target initvalue )  [private]
```

Method that allocates the container array. Deallocates if already allocated.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this,entries,tocopy*	

Definition at line 144 of file abstract_container_array.f90.

5.1.2.4 putvalue()

```
subroutine abstract_container_array_mod::putvalue (
             class(container_array), intent(inout) this,
             integer, intent(in) index,
             class(*), intent(in) value )  [private]
```

Method that stores a value on the requested index.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this,index,value*	

Definition at line 85 of file abstract_container_array.f90.

5.1.2.5 resizearray()

```
subroutine abstract_container_array_mod::resizearray (
            class(container_array), intent(inout) this,
            integer, intent(in) newsize,
            class(*), intent(in), optional, target initvalue )  [private]
```

Method that grows (adds empty space) or shrinks (discards the last entries) of the array. Use sparsely as this might get expensive for large array operations. Should think of a way to use move_alloc()

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this,newsize*	

Definition at line 116 of file abstract_container_array.f90.

5.1.3 Variable Documentation

5.1.3.1 mc

```
logical, target abstract_container_array_mod::mc = .true.  [private]
```

Definition at line 56 of file abstract_container_array.f90.

5.2 container_mod Module Reference

Module that defines an unlimited polymorphic container class and related methods. A container is a fundamental entity allowing to build data structures such as lists and arrays.

Data Types

- interface container

Functions/Subroutines

- class(∗) function, pointer getcontent (this)

  Method that returns a pointer to the values stored in the container.
- subroutine storecontent (this, to_store)

  Method that stores the provided value in the container using sourced allocation.
- subroutine printcontainer (this)

  Method to print the stored value. Only knows about instrinsic types, ignores (but warns) if other types are passed.
- class(container) function, pointer constructor (to_store)

  Container constructor, can be used with the 'container' name since it is defined as an interface.

5.2.1 Detailed Description

Module that defines an unlimited polymorphic container class and related methods. A container is a fundamental entity allowing to build data structures such as lists and arrays.

Author

Ricardo Birjukovs Canelas

5.2.2 Function/Subroutine Documentation

5.2.2.1 constructor()

```
class(container) function, pointer container_mod::constructor (
            class(*), intent(in) to_store )  [private]
```

Container constructor, can be used with the 'container' name since it is defined as an interface.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*to_store*	

Definition at line 109 of file container.f90.

5.2.2.2 getcontent()

```
class(*) function, pointer container_mod::getcontent (
            class(container), intent(in) this )  [private]
```

Method that returns a pointer to the values stored in the container.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this*	

Definition at line 62 of file container.f90.

5.2.2.3 printcontainer()

```
subroutine container_mod::printcontainer (
            class(container), intent(in) this )  [private]
```

Method to print the stored value. Only knows about instrinsic types, ignores (but warns) if other types are passed.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this*	

Definition at line 88 of file container.f90.

5.2.2.4 storecontent()

```
subroutine container_mod::storecontent (
            class(container), intent(inout) this,
            class(*), intent(in) to_store )  [private]
```

Method that stores the provided value in the container using sourced allocation.

Author

Ricardo Birjukovs Canelas - MARETEC

Parameters

in	*this,to_store*	

Definition at line 75 of file container.f90.

5.3 shape_array_mod Module Reference

Data Types

- type shapearray

Functions/Subroutines

- subroutine printshapearray (this)
- subroutine printshapeelement (this, index)

5.3.1 Function/Subroutine Documentation

5.3.1.1 printshapearray()

```
subroutine shape_array_mod::printshapearray (
             class(shapearray), intent(in) this )  [private]
```

Definition at line 35 of file shape_array.f90.

5.3.1.2 printshapeelement()

```
subroutine shape_array_mod::printshapeelement (
             class(shapearray), intent(in) this,
             integer, intent(in) index )  [private]
```

Definition at line 52 of file shape_array.f90.

5.4 types_mod Module Reference

Data Types

- type circle
- type shape

Functions/Subroutines

- subroutine printshape (this)
- subroutine printcircle (this)

5.4.1 Function/Subroutine Documentation

5.4.1.1 printcircle()

```
subroutine types_mod::printcircle (
            class(circle), intent(in) this )  [private]
```

Definition at line 45 of file types.f90.

5.4.1.2 printshape()

```
subroutine types_mod::printshape (
            class(shape), intent(in) this )  [private]
```

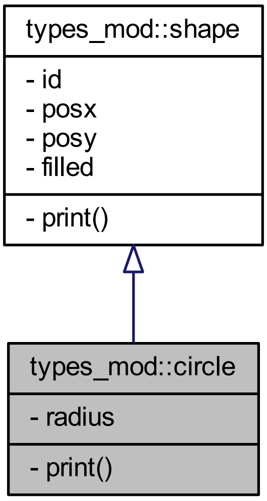
Definition at line 39 of file types.f90.

Chapter 6

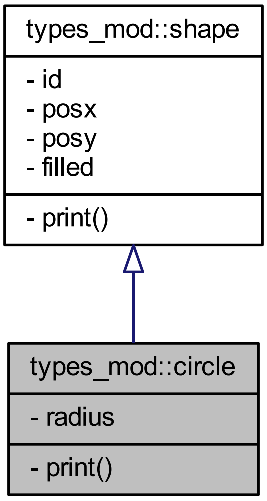
Data Type Documentation

6.1 types_mod::circle Type Reference

Inheritance diagram for types_mod::circle:



Collaboration diagram for types_mod::circle:



Private Member Functions

- procedure print =>printCircle

Private Attributes

- integer radius

6.1.1 Detailed Description

Definition at line 30 of file types.f90.

6.1.2 Member Function/Subroutine Documentation

6.1.2.1 print()

```
procedure types_mod::circle::print ( )  [private]
```

Definition at line 33 of file types.f90.

6.1.3 Member Data Documentation

6.1.3.1 radius

```
integer types_mod::circle::radius  [private]
```

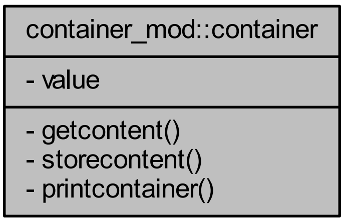
Definition at line 31 of file types.f90.

The documentation for this type was generated from the following file:

- C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/types.f90

6.2 container_mod::container Interface Reference

Collaboration diagram for container_mod::container:



Private Member Functions

- procedure getcontent

    returns stored content (pointer)
- procedure storecontent

    stores the provided values (sourced allocation)
- procedure printcontainer

    prints container contents (only primitive types implemented)

Private Attributes

- class(∗), pointer value => null()

    value stored in container

6.2.1 Detailed Description

Definition at line 40 of file container.f90.

6.2.2 Member Function/Subroutine Documentation

6.2.2.1 getcontent()

```
procedure container_mod::container::getcontent ( )  [private]
```

returns stored content (pointer)

Definition at line 44 of file container.f90.

6.2.2.2 printcontainer()

```
procedure container_mod::container::printcontainer ( )  [private]
```

prints container contents (only primitive types implemented)

Definition at line 46 of file container.f90.

6.2.2.3 storecontent()

```
procedure container_mod::container::storecontent ( )  [private]
```

stores the provided values (sourced allocation)

Definition at line 45 of file container.f90.

6.2.3 Member Data Documentation

6.2.3.1 value

```
class(*), pointer container_mod::container::value => null()  [private]
```

value stored in container

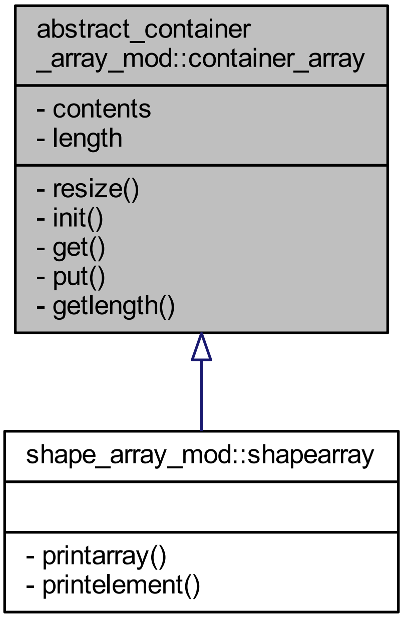
Definition at line 42 of file container.f90.

The documentation for this interface was generated from the following file:

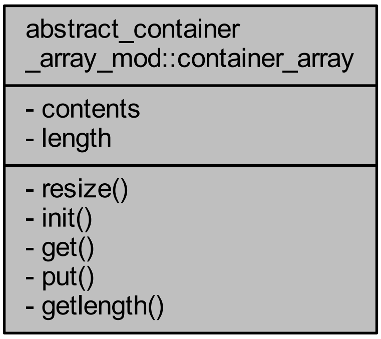
- C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/container.f90

6.3 abstract_container_array_mod::container_array Type Reference

Inheritance diagram for abstract_container_array_mod::container_array:



Collaboration diagram for abstract_container_array_mod::container_array:



Private Member Functions

- procedure resize => resizeArray

Grows (adds empty space) or shrinks (discards the last entries) of the array.

- procedure init => initArray

  Allocates the container array. Deallocates if already allocated.

- procedure, non_overridable get => getValue

  returns the requested entry (pointer)

- procedure, non_overridable put => putValue

  stores a value on the requested index

- procedure, non_overridable getlength

  returns the length of the array

Private Attributes

- class(container), dimension(:), allocatable contents

  Allocatable unlimited polymorphic container array.

- integer length

  Lenght of the array, for easy access.

6.3.1 Detailed Description

Definition at line 44 of file abstract_container_array.f90.

6.3.2 Member Function/Subroutine Documentation

6.3.2.1 get()

```
procedure, non_overridable abstract_container_array_mod::container_array::get ( )  [private]
```

returns the requested entry (pointer)

Definition at line 51 of file abstract_container_array.f90.

6.3.2.2 getlength()

```
procedure, non_overridable abstract_container_array_mod::container_array::getlength ( )  [private]
```

returns the length of the array

Definition at line 53 of file abstract_container_array.f90.

6.3.2.3 init()

```
procedure abstract_container_array_mod::container_array::init ( )  [private]
```

Allocates the container array. Deallocates if already allocated.

Definition at line 50 of file abstract_container_array.f90.

6.3.2.4 put()

```
procedure, non_overridable abstract_container_array_mod::container_array::put ( )  [private]
```

stores a value on the requested index

Definition at line 52 of file abstract_container_array.f90.

6.3.2.5 resize()

```
procedure abstract_container_array_mod::container_array::resize ( )  [private]
```

Grows (adds empty space) or shrinks (discards the last entries) of the array.

Definition at line 49 of file abstract_container_array.f90.

6.3.3 Member Data Documentation

6.3.3.1 contents

```
class(container), dimension(:), allocatable abstract_container_array_mod::container_array↩
::contents  [private]
```

Allocatable unlimited polymorphic container array.

Definition at line 46 of file abstract_container_array.f90.

6.3.3.2 length

```
integer abstract_container_array_mod::container_array::length  [private]
```

Lenght of the array, for easy access.

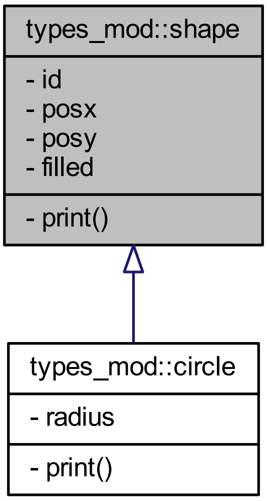
Definition at line 47 of file abstract_container_array.f90.

The documentation for this type was generated from the following file:

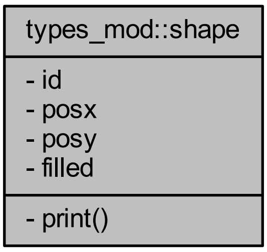
- C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/abstract_container_array.f90

6.4 types_mod::shape Type Reference

Inheritance diagram for types_mod::shape:



Collaboration diagram for types_mod::shape:



Private Member Functions

- procedure print =>printShape

Private Attributes

- integer id
- real posx
- real posy
- logical filled

6.4.1 Detailed Description

Definition at line 22 of file types.f90.

6.4.2 Member Function/Subroutine Documentation

6.4.2.1 print()

```
procedure types_mod::shape::print ( )  [private]
```

Definition at line 27 of file types.f90.

6.4.3 Member Data Documentation

6.4.3.1 filled

```
logical types_mod::shape::filled  [private]
```

Definition at line 25 of file types.f90.

6.4.3.2 id

```
integer types_mod::shape::id  [private]
```

Definition at line 23 of file types.f90.

6.4.3.3 posx

```
real types_mod::shape::posx  [private]
```

Definition at line 24 of file types.f90.

6.4.3.4 posy

```
real types_mod::shape::posy  [private]
```

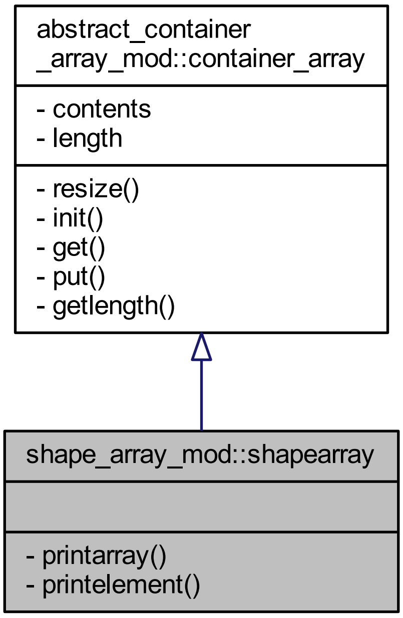
Definition at line 24 of file types.f90.

The documentation for this type was generated from the following file:

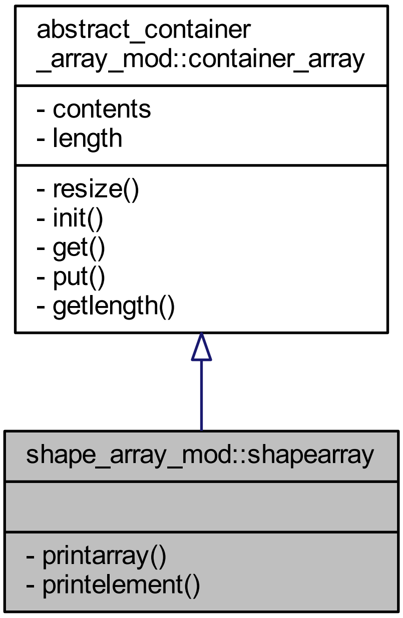
- C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/types.f90

6.5 shape_array_mod::shapearray Type Reference

Inheritance diagram for shape_array_mod::shapearray:



Collaboration diagram for shape_array_mod::shapearray:



Private Member Functions

- procedure printarray => printshapeArray
- procedure printelement => printshapeElement

6.5.1 Detailed Description

Definition at line 26 of file shape_array.f90.

6.5.2 Member Function/Subroutine Documentation

6.5.2.1 printarray()

```
procedure shape_array_mod::shapearray::printarray ( )  [private]
```

Definition at line 28 of file shape_array.f90.

6.5.2.2 printelement()

```
procedure shape_array_mod::shapearray::printelement ( )  [private]
```

Definition at line 29 of file shape_array.f90.

The documentation for this type was generated from the following file:

- C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/shape_array.f90

Chapter 7

File Documentation

7.1 C:/Users/administrator/Documents/GitHub/FUPRA/src/app/TestsFUPRA.f90 File Reference

Functions/Subroutines

- program testsfupra

7.1.1 Function/Subroutine Documentation

7.1.1.1 testsfupra()

```
program testsfupra ( )
```

Definition at line 18 of file TestsFUPRA.f90.

7.2 C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/abstract_container_↩ array.f90 File Reference

Data Types

- type abstract_container_array_mod::container_array

Modules

- module abstract_container_array_mod

  Module that defines an unlimited polymorphic container class and related methods. A container is a fundamental entity allowing to build data structures such as lists and arrays. This is an abstract type, so a derived type must be defined for any specific contents that may be required. Those derived types should provide type-specific methods that require type-guards, such as printing.

Functions/Subroutines

- class(∗) function, pointer abstract_container_array_mod::getvalue (this, index)

  Method that returns returns the requested entry (pointer)
- subroutine abstract_container_array_mod::putvalue (this, index, value)

  Method that stores a value on the requested index.
- integer function abstract_container_array_mod::getlength (this)

  Method that returns the length of the array.
- subroutine abstract_container_array_mod::resizearray (this, newsize, initvalue)

  Method that grows (adds empty space) or shrinks (discards the last entries) of the array. Use sparsely as this might get expensive for large array operations. Should think of a way to use move_alloc()
- subroutine abstract_container_array_mod::initarray (this, entries, source, initvalue)

  Method that allocates the container array. Deallocates if already allocated.

Variables

- logical, target abstract_container_array_mod::mc = .true.

7.3 C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/container.f90 File Reference

Data Types

- interface container_mod::container
- interface container_mod::container

Modules

- module container_mod

  Module that defines an unlimited polymorphic container class and related methods. A container is a fundamental entity allowing to build data structures such as lists and arrays.

Functions/Subroutines

- class(∗) function, pointer container_mod::getcontent (this)

  Method that returns a pointer to the values stored in the container.
- subroutine container_mod::storecontent (this, to_store)

  Method that stores the provided value in the container using sourced allocation.
- subroutine container_mod::printcontainer (this)

  Method to print the stored value. Only knows about instrinsic types, ignores (but warns) if other types are passed.
- class(container) function, pointer container_mod::constructor (to_store)

  Container constructor, can be used with the 'container' name since it is defined as an interface.

7.4 C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/shape_array.f90 File Reference

Data Types

- type shape_array_mod::shapearray

Modules

- module shape_array_mod

Functions/Subroutines

- subroutine shape_array_mod::printshapearray (this)
- subroutine shape_array_mod::printshapeelement (this, index)

7.5 C:/Users/administrator/Documents/GitHub/FUPRA/src/lib/types.f90 File Reference

Data Types

- type types_mod::shape
- type types_mod::circle

Modules

- module types_mod

Functions/Subroutines

- subroutine types_mod::printshape (this)
- subroutine types_mod::printcircle (this)