

---

**AIMMS Function Reference - GMPCoefficient Procedures and Functions**

This file contains only one chapter of the book. For a free download of the complete book in pdf format, please visit [www.aimms.com](http://www.aimms.com)

**Part II**

---

**The GMP library**

# The GMP library

Through the GMP library you have direct access to mathematical program instances generated by AIMMS, allowing you to implement advanced algorithms in an efficient manner. The GMP routines can also be used for nonlinear models, unless specified otherwise. All procedures and functions in the GMP library are part of the GMP namespace in AIMMS. This namespace is subdivided into the following functional namespaces:

- Procedures and functions in the GMP::Coefficient namespace
- Procedures and functions in the GMP::Column namespace
- Procedures and functions in the GMP::Event namespace
- Procedures and functions in the GMP::Instance namespace
- Procedures and functions in the GMP::Linearization namespace
- Procedures and functions in the GMP::ProgressWindow namespace
- Procedures and functions in the GMP::QuadraticCoefficient namespace
- Procedures and functions in the GMP::Robust namespace
- Procedures and functions in the GMP::Row namespace
- Procedures and functions in the GMP::Solution namespace
- Procedures and functions in the GMP::Solver namespace
- Procedures and functions in the GMP::SolverSession namespace
- Procedures and functions in the GMP::Stochastic namespace
- Procedures and functions in the GMP::Tuning namespace

## GMP::Coefficient Procedures and Functions

AIMMS supports the following procedures and functions for modifying the coefficient matrix associated with a generated mathematical program instance:

- `GMP::Coefficient::Get`
- `GMP::Coefficient::GetQuadratic`
- `GMP::Coefficient::Set`
- `GMP::Coefficient::SetQuadratic`

---

## GMP::Coefficient::Get

The function `GMP::Coefficient::Get` retrieves a (linear) coefficient in a generated mathematical program.

```
GMP::Coefficient::Get(
  GMP,          ! (input) a generated mathematical program
  row,          ! (input) a scalar reference
  column       ! (input) a scalar reference
)
```

### Arguments:

*GMP*

An element in `AllGeneratedMathematicalPrograms`.

*row*

A scalar reference to an existing row in the model.

*column*

A scalar reference to an existing column in the model.

### Return value:

The value of the specified coefficient in the generated mathematical program.

### Remarks:

In case the generated mathematical program is nonlinear, this function will return 0 if the column is part of a nonlinear term in the row. However, if the row is pure quadratic then this function will return the linear coefficient value for a quadratic column.

### Examples:

Assume `GMP` contains a constraint `e1` with definition  $2*x1 + 3*x2 + x2^3$ . Then `GMP::Coefficient::Get(GMP,e1,x1)` will return 2. Because column `x2` is part of the nonlinear term  $x2^3$ , `GMP::Coefficient::Get(GMP,e1,x2)` will return 0.

### See also:

The routines `GMP::Coefficient::Set` and `GMP::QuadraticCoefficient::Get`.

---

## GMP::Coefficient::GetQuadratic

The function `GMP::Coefficient::GetQuadratic` retrieves the value of a quadratic product between two columns in a generated mathematical program.

```
GMP::Coefficient::GetQuadratic(
  GMP,           ! (input) a generated mathematical program
  column1,      ! (input) a scalar reference
  column2       ! (input) a scalar reference
)
```

### Arguments:

*GMP*

An element in `AllGeneratedMathematicalPrograms`.

*column1*

A scalar reference to an existing column in the model.

*column2*

A scalar reference to an existing column in the model.

### Return value:

The value of the specified quadratic term in the generated mathematical program.

### Remarks:

- If *column1* equals *column2* then AIMMS multiplies the quadratic coefficient by 2 before it is returned by this function.
- This function operates on the objective. To get a quadratic coefficient in a row the function `GMP::QuadraticCoefficient::Get` should be used.

### See also:

The routines `GMP::Coefficient::SetQuadratic` and `GMP::QuadraticCoefficient::Get`.

---

## GMP::Coefficient::Set

The procedure `GMP::Coefficient::Set` sets the value of a (linear) coefficient in a generated mathematical program.

```
GMP::Coefficient::Set(
  GMP,          ! (input) a generated mathematical program
  row,          ! (input) a scalar reference
  column,       ! (input) a scalar reference
  value         ! (input) a scalar numerical value
)
```

### Arguments:

*GMP*

An element in `AllGeneratedMathematicalPrograms`.

*row*

A scalar reference to an existing row in the model.

*column*

A scalar reference to an existing column in the model.

*value*

A scalar numerical value indicating the value for the coefficient.

### Return value:

The procedure returns 1 on success, or 0 otherwise.

### Remarks:

- This procedure cannot be used if the column refers to the objective variable.
- In case the generated mathematical program is nonlinear, this procedure will fail if the column is part of a nonlinear term in the row. However, if the row is pure quadratic then this procedure can be used to set the linear coefficient value for a quadratic column.

### See also:

The routines `GMP::Coefficient::Get` and `GMP::QuadraticCoefficient::Set`.

---

## GMP::Coefficient::SetQuadratic

The procedure `GMP::Coefficient::SetQuadratic` sets the value of a quadratic product between two columns in a generated mathematical program.

```
GMP::Coefficient::SetQuadratic(
  GMP,          ! (input) a generated mathematical program
  column1,     ! (input) a scalar value
  column2,     ! (input) a scalar value
  value        | (input) a scalar numerical value
)
```

### Arguments:

*GMP*

An element in `AllGeneratedMathematicalPrograms`.

*column1*

A scalar reference to an existing column in the model.

*column2*

A scalar reference to an existing column in the model.

*value*

A scalar numerical value indicating the value for the quadratic term.

### Return value:

The procedure returns 1 on success, or 0 otherwise.

### Remarks:

- If *column1* equals *column2* then `AIMMS` multiplies the quadratic coefficient by 0.5 before it is stored (and passed to the solver).
- This procedure operates on the objective. To set a quadratic coefficient in a row the procedure `GMP::QuadraticCoefficient::Set` should be used.

### See also:

The routines `GMP::Coefficient::GetQuadratic` and `GMP::QuadraticCoefficient::Set`.