AIMMS Function Reference - Financial Functions - Investments

Financial Functions - Investments

Investments and loans

When dealing with investments or loans, several cash flows are scheduled within a certain time frame, such as the

- present value (the value at the beginning of the scheduled time frame),
- future value (the value at the end of the scheduled time frame), and
- periodic payments during the scheduled time frame.

AIMMS provides several functions to calculate each of these cash flows (or the interest rate used) in the presence of all others.

Constant payments

Investments and loans with constant, periodic payments and a constant interest rate are special. When the payments are annual, such an investment is called an annuity. The constant payments of these investments consist of a principal and an interest payment. The principal payment will generally increase in time whereas the interest payment will decrease in time. Two different types of investments with constant payments and interest rates can be distinguished:

- The first type, also referred as type 0, has payments that are made at the end of each period.
- The second type, type 1, has payments that are made at the beginning of each period. This type has no interest payment at the beginning of the first period, but does have an extra period, after the last periodic payment, with an interest payment over the last period and an inverse principal payment.

Equations

Cash flows can be either positive or negative, where a positive payment indicates that you are receiving this payment. Taking the interest into account, the total value of an investment must be equal to zero after all cash flows have occurred. For example, a positive present value and positive payments will lead to a negative future value: your debt has grown. The following equation expresses the relation between all the cash flows that take place

$$v_p(1+r)^N + p\sum_{i=1}^{N}(1+r)^{i-1+T} + v_f = 0$$

where v_p is the present value, v_f is the future value, p is the constant periodic payment, r is the constant interest rate and T is the investment type as discussed above.

AIMMS supports the following investment functions with constant, periodic payments:

- InvestmentConstantPresentValue
- InvestmentConstantFutureValue
- InvestmentConstantPeriodicPayment
- InvestmentConstantInterestPayment
- InvestmentConstantPrincipalPayment
- InvestmentConstantCumulativeInterestPayment
- InvestmentConstantCumulativePrincipalPayment
- InvestmentConstantNumberPeriods
- InvestmentConstantRateAll
- InvestmentConstantRate

Variable payments

When the cash flows are variable (i.e. not constant), take place at irregular intervals, or when the interest rate varies over time, it still possible to compute present values, future values, and the internal rate of return, i.e. the rate received for an investment consisting of payments and income.

AIMMS supports the following investment functions for variable cash flows:

- InvestmentVariablePresentValue
- InvestmentVariablePresentValueInPeriodic
- InvestmentSingleFutureValue
- InvestmentVariableInternalRateReturnAll
- InvestmentVariableInternalRateReturn
- InvestmentVariableInternalRateReturnInPeriodicAll
- InvestmentVariableInternalRateReturnInPeriodic
- InvestmentVariableInternalRateReturnModified

InvestmentConstantPresentValue

The function `InvestmentConstantPresentValue` returns the present value of an investment based on periodic, constant payments and a constant interest rate.

```
InvestmentConstantPresentValue(
    FutureValue,            ! (input) numerical expression
    Payment,                ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

Payment
: The periodic payment for the investment. *Payment* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantPresentValue` returns the total amount that a series of future payments is worth at this moment.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *FutureValue*, *Payment* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantPresentValue` is similar to the Excel function `PV`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantFutureValue

The function `InvestmentConstantFutureValue` returns the future value of an investment based on periodic, constant payments and a constant interest rate.

```
InvestmentConstantFutureValue(
    PresentValue,           ! (input) numerical expression
    Payment,                ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

Payment
: The periodic payment for the investment. *Payment* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantFutureValue` returns the cash balance you want to attain after the last payment is made.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue*, *Payment* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantFutureValue` is similar to the Excel function `FV`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantPeriodicPayment

The function `InvestmentConstantPeriodicPayment` returns the periodic payment for an investment based on periodic, constant payments and a constant interest rate.

```
InvestmentConstantPeriodicPayment(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantPeriodicPayment` returns the periodic payment for the investment.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue*, *FutureValue* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantPeriodicPayment` is similar to the Excel function `PMT`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantInterestPayment

The function `InvestmentConstantInterestPayment` returns the interest payment of the specified period for an investment based on periodic, constant payments and a constant interest rate. Every periodic payment can be divided in two parts: an interest payment and a principal repayment.

```
InvestmentConstantInterestPayment(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    Period                  ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

Period
: The period for which you want to compute the interest payment. *Period* must be an integer in the range $\{1, \mathit{NumberPeriods} + \mathit{Type}\}$. When $\mathit{Type} = 1$, the extra period is to account the interest over the former period.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. $\mathit{Type} = 0$: Payments are due at the end of each period. $\mathit{Type} = 1$: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantInterestPayment` returns the interest payment for the specified period.

Equation:

The interest payment i_i in period i is computed through the equation

$$i_i = -v_p r(1+r)^{i-1-T} - p\left(\left((1+r)^{i-1-T} - 1\right)(1+r)^T + rT\right)$$

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue*, *FutureValue* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantInterestPayment` is similar to the Excel function `IPMT`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantPrincipalPayment

The function `InvestmentConstantPrincipalPayment` returns the principal payment of the specified period for an investment based on periodic, constant payments and a constant interest rate. Every periodic payment can be divided in two parts: an interest payment and a principal payment.

```
InvestmentConstantPrincipalPayment(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    Period                  ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

Period
: The period for which you want to compute the interest payment. *Period* must be an integer in the range $\{1, \mathit{NumberPeriods} + \mathit{Type}\}$. When $\mathit{Type} = 1$, the extra period is to account the interest over the former period.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. $\mathit{Type} = 0$: Payments are due at the end of each period. $\mathit{Type} = 1$: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantPrincipalPayment` returns the principal payment for the specified period.

Equation:

The principal payment p_i in period i follows from the relation

$$p_i = p - i_i$$

where i_i is the interest payment in period i.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue*, *FutureValue* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantPrincipalPayment` is similar to the Excel function `PPMT`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantCumulativeInterestPayment

The function `InvestmentConstantCumulativeInterestPayment` returns the cumulative interest payment for the specified interval for an investment based on periodic, constant payments and a constant interest rate. Every periodic payment can be divided in two parts: an interest payment and a principal payment.

```
InvestmentConstantCumulativeInterestPayment(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    StartPeriod,            ! (input) numerical expression
    EndPeriod,              ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

StartPeriod
: The starting period of the interval for which you want to compute the cumulative interest payment. *StartPeriod* must be an integer in the range $\{1, NumberPeriods\}$.

EndPeriod
: The ending period of the interval for which you want to compute the cumulative interest payment. *EndPeriod* must be an integer in the range $\{StartPeriod, NumberPeriods\}$.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantCumulativeInterestPayment` returns the sum of the interest payments for the periods in the specified interval.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue*, *FutureValue* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantCumulativeInterestPayment` is similar to the Excel function `CUMIPMT`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantCumulativePrincipalPayment

The function `InvestmentConstantCumulativePrincipalPayment` returns the cumulative principal payment for the specified interval for an investment based on periodic, constant payments and a constant interest rate. Every periodic payment can be divided in two parts: an interest payment and a principal payment.

```
InvestmentConstantCumulativePrincipalPayment(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    StartPeriod,            ! (input) numerical expression
    EndPeriod,              ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

StartPeriod
: The starting period of the interval for which you want to compute the cumulative interest payment. *StartPeriod* must be an integer in the range $\{1, NumberPeriods\}$.

EndPeriod
: The ending period of the interval for which you want to compute the cumulative interest payment. *EndPeriod* must be an integer in the range $\{StartPeriod, NumberPeriods\}$.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantCumulativePrincipalPayment` returns the sum of the principal payments for the periods in the specified interval.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue*, *FutureValue* and *InterestRate* can be used as a variable.
- The function `InvestmentConstantCumulativePrincipalPayment` is similar to the Excel function `CUMPRINC`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantNumberPeriods

The function `InvestmentConstantNumberPeriods` returns the number of periods for an investment based on periodic, constant payments and a constant interest rate.

```
InvestmentConstantNumberPeriods(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    Payment,                ! (input) numerical expression
    InterestRate,           ! (input) numerical expression
    Type                    ! (input) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

Payment
: The value of the periodic payment for the investment. Payment must be a real number. *Payment* and *InterestRate* cannot both be 0.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Return value:

The function `InvestmentConstantNumberPeriods` returns the number of periods for an investment based on periodic, constant payments and a constant interest rate.

Remarks:

The function `InvestmentConstantNumberPeriods` is similar to the Excel function `NPER`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantRateAll

The procedure `InvestmentConstantRateAll` returns the interest rate(s) for an investment based on periodic, constant payments and a constant interest rate.

```
InvestmentConstantRateAll(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    Payment,                ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    Type,                   ! (input) numerical expression
    Mode,                   ! (input) numerical expression
    NumberSolutions,        ! (output) numerical expression
    Solutions               ! (output) one-dimensional parameter
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

Payment
: The periodic payment for the investment. *Payment* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

Mode
: Indicates whether all the solutions need to be found or just one. *Mode* = 0: the search for solutions stops after one solution is found. *Mode* = 1: the search for solutions continues till all solutions are found.

NumberSolutions
: The number of solutions found. If *Mode* = 0 *NumberSolutions* will always be 1.

Solutions
: There is not always a unique solution for *InterestRate*. Dependent on *Mode* one solution or all the solutions will be given. Solutions smaller than −1 are not supposed to be relevant, so the search for solutions is limited to the area greater than −1.

Remarks:

- When you want to use this procedure in an objective function or constraint you have to use `InvestmentConstantRate`.
- The function `InvestmentConstantRateAll` is similar to the Excel function `RATE`.

See also:

General equations for investments with constant, periodic payments.

InvestmentConstantRate

The function InvestmentConstantRate returns the interest rate for an investment based on periodic, constant payments and a constant interest rate. This function uses the procedure InvestmentConstantRateAll to determine all possible interest rates and returns the interest rate that is within the specified bounds.

```
InvestmentConstantRate(
    PresentValue,           ! (input) numerical expression
    FutureValue,            ! (input) numerical expression
    Payment,                ! (input) numerical expression
    NumberPeriods,          ! (input) numerical expression
    Type,                   ! (input) numerical expression
    [LowerBound,]           ! (optional) numerical expression
    [UpperBound,]           ! (optional) numerical expression
    [Error]                 ! (optional) numerical expression
    )
```

Arguments:

PresentValue
: The total amount that a series of future payments is worth at this moment. *PresentValue* must be a real number.

FutureValue
: The cash balance you want to attain after the last payment is made. *FutureValue* must be a real number.

Payment
: The periodic payment for the investment. *Payment* must be a real number.

NumberPeriods
: The total number of payment periods for the investment. *NumberPeriods* must be a positive integer.

Type
: Indicates when payments are due. *Type* = 0: Payments are due at the end of each period. *Type* = 1: Payments are due at the beginning of each period.

LowerBound
: Indicates a minimum for the interest rate to be accepted by this function. The default is −1.

UpperBound
: Indicates a maximum for the interest rate to be accepted by this function. The default is 5.

Error
: Indicates whether AIMMS should give an error if multiple solutions are found that satisfy the bounds. *Error* = 0: if multiple solutions are

found, return the solution with the smallest absolute value. *Error* = 1: if multiple solutions are found, return an error message. The default is 0.

Return value:

The function `InvestmentConstantRate` returns the interest rate for an investment based on periodic, constant payments and a constant interest rate.

Remarks:

- The function `InvestmentConstantRate` can be used in an objective function or constraint. The input parameters *PresentValue*, *FutureValue* and *Payment* can be used as variables.
- The function `InvestmentConstantRate` is similar to the Excel function `RATE`.

See also:

General equations for investments with constant, periodic payments.

InvestmentVariablePresentValue

The function `InvestmentVariablePresentValue` returns the net present value for an investment based on a series of periodic cash flows at the end of the periods and a constant interest rate.

```
InvestmentVariablePresentValue(
    Value,                  ! (input) one-dimensional numerical parameter
    InterestRate            ! (input) numerical expression
    )
```

Arguments:

Value
: The periodic payments (positive or negative), which must be equally spaced in time and occur at the end of each period. The order of the payments in *Value* must be the same as the order in which the cash flows occur. *Value* is an one dimensional parameter of real numbers. *Value* should contain at least one nonzero number. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Return value:

The function `InvestmentVariablePresentValue` returns the net present value of an investment, which is the total value of all the future cash flows at the beginning of the first period.

Equation:

The net present value v_p is computed through the equation

$$v_p = \sum_{i=1}^{n} \frac{p_i}{(1+r)^i}$$

where p_i are the (variable) periodic payments, and r is the (constant) interest rate.

Remarks:

- When all payments are constant, the net present value computed here is equal to the negative value of the present value computed by the function `InvestmentConstantPresentValue` with the future value set to 0.0.
- This function can be used in an objective function or constraint and the input parameters *Value* and *InterestRate* can be used as a variable.

- The function `InvestmentVariablePresentValue` is similar to the Excel function `NPV`.

See also:

The function `InvestmentConstantPresentValue`.

InvestmentVariablePresentValueInPeriodic

The function `InvestmentVariablePresentValueInPeriodic` returns the net present value on the date of the first cash flow for an investment based on a series of in-periodic cash flows and a constant interest rate.

```
InvestmentVariablePresentValueInPeriodic(
    Value,                  ! (input) one-dimensional numerical expression
    Date,                   ! (input) one-dimensional string expression
    InterestRate,           ! (input) numerical expression
    [Basis]                 ! (optional) numerical expression
    )
```

Arguments:

Value
: The payments (positive or negative). *Value* is an one-dimensional parameter of real numbers. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts. *Value* must contain at least one positive and at least one negative number.

Date
: The dates on which the payments occur. *Date* and *Value* must have the same order. *Date* is an one-dimensional parameter of dates given in a date format. The first payment date indicates the beginning of the schedule of payments. All other dates must be later than this date, but they may occur in any order. *Date* should contain as many dates as the number of values given by *Value*.

InterestRate
: The interest rate per period for the investment. *InterestRate* must be a numerical expression in the range $(-1, 1)$.

Basis
: The day-count basis method to be used. The default is 1.

Return value:

The function `InvestmentVariablePresentValueInPeriodic` returns the net present value of an investment, which is the total value of all the future cash flows at this moment.

Equation:

The net present value v_p is computed through the equation

$$v_p = \sum_{i=1}^{n} \frac{p_i}{(1+r)^{f_i}}$$

where p_i are the periodic payments, r is the (constant) interest rate, and f_i is the difference between date i and the first date (so, $f_1 = 0$), according to the selected day-count basis method.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *Value* and *InterestRate* can be used as a variable.
- The function `InvestmentVariablePresentValueInPeriodic` is similar to the Excel function `XNPV`.

See also:

Day count basis methods.

InvestmentSingleFutureValue

The function `InvestmentSingleFutureValue` returns the future value, the cash balance, of a payment made at this moment, present value, with periodic interest rates.

```
InvestmentSingleFutureValue(
    PresentValue,           ! (input) numerical expression
    PeriodicRate            ! (input) one-dimensional numerical expression
    )
```

Arguments:

PresentValue
: Payment made at the start of the first period. *PresentValue* must be a real number. If *PresentValue* is a negative number it represents an outgoing amount and when it is a positive number it represents an incoming amount.

PeriodicRate
: Interest rates which differ per period. *PeriodicRate* is a one-dimensional parameter, which should contain at least one nonzero number. The periods must be equally spaced in time and the interest rates must be ordered.

Return value:

The function `InvestmentSingleFutureValue` returns the future value of the present value, using the periodic interest rates.

Equation:

The future value v_f is computed through the equation

$$v_f = v_p \prod_{i=1}^{n} (1 + r_i)$$

where v_p is the present value, and r_i the variable, periodic interest rates.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *PresentValue* and *PeriodicRate* can be used as a variable.
- The function `InvestmentSingleFutureValue` is similar to the Excel function `FVSCHEDULE`.

InvestmentVariableInternalRateReturnAll

The procedure `InvestmentVariableInternalRateReturnAll` returns the internal rate of return for an investment based on a series of periodic cash flows. The internal rate of return is the rate received for an investment consisting of payments (negative values) and income (positive values).

```
InvestmentVariableInternalRateReturnAll(
    Value,                  ! (input) one-dimensional numerical expression
    Mode,                   ! (input) numerical expression
    NumberSolutions,        ! (output) numerical expression
    IRR                     ! (output) one-dimensional numerical expression
    )
```

Arguments:

Value
: The periodic payments (positive or negative), which must be equally spaced in time. The order of the payments in *Value* must be the same as the order in which the cash flows occur. *Value* is an one dimensional parameter of real numbers. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts. em Value must contain at least one positive and at least one negative number.

Mode
: Indicates whether all the solutions need to be found or just one. *Mode* = 0: the search for solutions stops after one solution is found. *Mode* = 1: the search for solutions continues till all solutions are found.

NumberSolutions
: The number of solutions found. When *Mode* = 0 the *NumberSolutions* will be 1.

IRR
: The internal rate of return for the investment. There is not always a unique solution for *IRR*. Dependent on *Mode* one solution or all the solutions will be given. Solutions smaller than -1 are not supposed to be relevant, so the search for solutions is limited to the area greater than -1.

Equation:

The internal rate of return r is a solution of the equation

$$\sum_{i=1}^{n} \frac{p_i}{(1+r)^i} = 0$$

where p_i are the periodic payments.

Remarks:

- The internal rate of return is the interest rate at which the investment has a zero net present value.
- When you want to use this procedure in an objective function or constraint you have to use *InvestmentVariableInternalRateReturn*.
- The function `InvestmentVariableInternalRateReturnAll` is similar to the Excel function `IRR`.

See also:

The functions `InvestmentVariableInternalRateReturn`, `InvestmentVariableInternalRateReturnInPeriodic`.

InvestmentVariableInternalRateReturn

The function InvestmentVariableInternalRateReturn returns the internal rate of return for an investment based on a series of periodic cash flows. The internal rate of return is the rate received for an investment. This function uses the procedure *InvestmentVariableInternalRateReturnAll* to determine all possible internal rates and returns the internal rate that is within the specified bounds.

```
InvestmentVariableInternalRateReturn(
    Value,                  ! (input) one-dimensional numerical expression
    [LowerBound,]           ! (optional) numerical expression
    [UpperBound,]           ! (optional) numerical expression
    [Error]                 ! (optional) numerical expression
    )
```

Arguments:

Value
: The periodic payments (positive or negative), which must be equally spaced in time. The order of the payments in *Value* must be the same as the order in which the cash flows occur. *Value* is an one dimensional parameter of real numbers. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts. em Value must contain at least one positive and at least one negative number.

LowerBound
: Indicates a minimum for the internal rate to be accepted by this function. The default is −1.

UpperBound
: Indicates a maximum for the internal rate to be accepted by this function. The default is 5.

Error
: Indicates whether AIMMS should give an error if multiple solutions are found that satisfy the bounds. *Error* = 0: if multiple solutions are found, return the solution with the smallest absolute value. *Error* = 1: if multiple solutions are found, return an error message. The default is 0.

Return value:

The function InvestmentVariableInternalRateReturn returns the internal rate of return for an investment based on a series of periodic cash flows. The internal rate of return is the rate received for an investment.

Remarks:

- The function `InvestmentVariableInternalRateReturn` can be used in an objective function or constraint. The input parameter *Value* can be used as a variable.
- The function `InvestmentVariableInternalRateReturn` is similar to the Excel function `IRR`.

See also:

The functions `InvestmentVariableInternalRateReturnAll`, `InvestmentVariableInternalRateReturnInPeriodic`.

InvestmentVariableInternalRateReturnInPeriodicAll

The procedure `InvestmentVariableInternalRateReturnInPeriodicAll` returns the internal rate of return for an investment based on a series of in-periodic cash flows. The internal rate of return is the interest rate received for an investment.

```
InvestmentVariableInternalRateReturnInPeriodicAll(
    Value,                  ! (input) one-dimensional numerical expression
    Date,                   ! (input) one-dimensional string expression
    Mode,                   ! (input) numerical expression
    IRR,                    ! (output) one-dimensional numerical expression
    NumberSolutions,        ! (output) numerical expression
    [Basis]                 ! (optional) numerical expression
    )
```

Arguments:

Value

The payments (positive or negative). *Value* is an one-dimensional parameter of real numbers. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts. *Value* must contain at least one positive and at least one negative number.

Date

The dates on which the payments occur. *Date* and *Value* must have the same order. *Date* is an one-dimensional parameter of dates given in a date format. The first payment date indicates the beginning of the schedule of payments. All other dates must be later than this date, but they may occur in any order. *Date* should contain as many dates as the number of values given by *Value*.

Mode

Indicates whether all the solutions need to be found or just one. *Mode* = 0: the search for solutions stops after one solution is found. *Mode* = 1: the search for solutions continues till all solutions are found.

IRR

The internal rate of return for the investment. There is not always a unique solution for *IRR*. Dependent on *Mode* one solution or all the solutions will be given. Solutions smaller than -1 are not supposed to be relevant, so the search for solutions is limited to the area greater than -1.

NumberSolutions

The number of solutions found. When *Mode* = 0 the *NumberSolutions* will be 1.

Basis

The day-count basis method to be used. The default is 1.

Equation:

The internal rate of return r is a solution of the equation

$$\sum_{i=1}^{n} \frac{p_i}{(1+r)^{f_i}} = 0$$

where p_i are the periodic payments, and f_i is the difference between date i and the first date (so, $f_1 = 0$), according to the selected day-count basis method.

Remarks:

- When you want to use the procedure in an objective function or constraint you have to use `InvestmentVariableInternalRateReturnInPeriodic`.
- The procedure `InvestmentVariableInternalRateReturnInPeriodicAll` is similar to the Excel function `XIRR`.

See also:

The functions `InvestmentVariableInternalRateReturn`, `InvestmentVariableInternalRateReturnInPeriodic`. Day count basis methods.

InvestmentVariableInternalRateReturnInPeriodic

The function InvestmentVariableInternalRateReturnInPeriodic returns the internal rate of return for an investment based on a series of in-periodic cash flows. The internal rate of return is the interest rate received for an investment. This function uses the procedure InvestmentVariableInternalRateReturnInPeriodicAll to determine all possible internal rates and returns the internal rate that is within the specified bounds.

```
InvestmentVariableInternalRateReturnInPeriodic(
    Value,                  ! (input) one-dimensional numerical expression
    Date,                   ! (input) one-dimensional string expression
    [Basis,]                ! (optional) numerical expression
    [LowerBound,]           ! (optional) numerical expression
    [UpperBound,]           ! (optional) numerical expression
    [Error]                 ! (optional) numerical expression
    )
```

Arguments:

Value

The periodic payments (positive or negative), which must be equally spaced in time. The order of the payments in *Value* must be the same as the order in which the cash flows occur. *Value* is an one dimensional parameter of real numbers. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts. em Value must contain at least one positive and at least one negative number.

Date

The dates on which the payments occur. *Date* and *Value* must have the same order. *Date* is an one-dimensional parameter of dates given in a date format. The first payment date indicates the beginning of the schedule of payments. All other dates must be later than this date, but they may occur in any order. *Date* should contain as many dates as the number of values given by *Value*.

Basis

The day-count basis method to be used. The default is 1.

LowerBound

Indicates a minimum for the internal rate to be accepted by this function. The default is −1.

UpperBound

Indicates a maximum for the internal rate to be accepted by this function. The default is 5.

Error

Indicates whether AIMMS should give an error if multiple solutions are found that satisfy the bounds. *Error* = 0: if multiple solutions are

found, return the solution with the smallest absolute value. *Error* = 1: if multiple solutions are found, return an error message. The default is 0.

Return value:

The function `InvestmentVariableInternalRateReturnInPeriodic` returns the internal rate of return for an investment based on a series of in-periodic cash flows. The internal rate of return is the interest rate received for an investment.

Remarks:

- The function `InvestmentVariableInternalRateReturnInPeriodic` can be used in an objective function or constraint. The input parameter *Value* can be used as a variable.
- The function `InvestmentVariableInternalRateReturnInPeriodic` is similar to the Excel function `XIRR`.

See also:

The functions `InvestmentVariableInternalRateReturn`, `InvestmentVariableInternalRateReturnInPeriodicAll`. Day count basis methods.

InvestmentVariableInternalRateReturnModified

The function `InvestmentVariableInternalRateReturnModified` returns the modified internal rate of return for an investment based on a series of periodic cash flows. It considers both the cost made for the investment and the interest received on the reinvestment of cash flows.

```
InvestmentVariableInternalRateReturnModified(
    Value,                  ! (input) one-dimensional numerical expression
    FinanceRate,            ! (input) numerical expression
    ReinvestRate            ! (input) numerical expression
    )
```

Arguments:

Value
: The periodic payments (positive or negative), which must be equally spaced in time. The order of the payments in *Value* must be the same as the order in which the cash flows occur. *Value* is an one dimensional parameter of real numbers. *Value* given by positive numbers represent incoming amounts and *Value* given by negative numbers represent outgoing amounts. *Value* must contain at least one positive and at least one negative number.

FinanceRate
: Interest rate you pay on money used in negative cash flows. *FinanceRate* must be a numerical expression in the range $[-1, \infty)$.

ReinvestRate
: Interest rate you receive on the positive cash flows as you reinvest them. *ReinvestRate* must be a numerical expression in the range $[-1, \infty)$.

Return value:

The function `InvestmentVariableInternalRateReturnModified` returns the modified internal rate of return for the investment.

Equation:

The internal rate of return r is the solution of the equation

$$(1+r)^{n-1} = -\frac{\mathrm{NPV}(v^+, r_r)(1+r_r)^n}{\mathrm{NPV}(v^-, r_f)(1+r_f)}$$

where n is the number of periods considered, $v_i = v_i^+ - v_i^-$ (with $v_i^+, v_i^- \geq 0$), r_f the finance rate, r_r the reinvestment rate, and NPV the function `InvestmentVariablePresentValue`.

Remarks:

- This function can be used in an objective function or constraint and the input parameters *Value*, *FinanceRate* and *ReinvestRate* can be used as a variable.
- There should be at least one negative and one negative *Value*.
- The function `InvestmentVariableInternalRateReturnModified` is similar to the Excel function `MIRR`.

See also:

The function `InvestmentVariableInternalRateReturn`.