Library 1672: Financial Mathematics

Content:

1. Disclaimer & Copyright

This program is for your private use only and is provided "as is".
This software is not sold, only the right for using it is granted. Using this software is only allowed on the calculator the software has been licensed for.
This program has been tested but may contain errors. I'm making no warranty of any kind with regard to this software, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. I shall not be liable for any errors or for incidental or consequential damages in connection with the furnishing, performance, or use of this software. Suggestions, criticism and/or improvement suggestions can be send to the author at Software49g@gmx.de.

2. Credits

Thanks to ACO for the HP 49G, Pieter Kuyck for explaining IfMain, Wolfgang Rautenberg for OT49, Eduardo M. Kalinowski for the excellent book "Programming in System RPL", Mika Heiskanen for BZ, William G. Graves for Debug4x and various post from different authors in comp.sys.hp48, without them this program wouldn't been written.
The code for changing the speed is based on routines by Al Borowski and the ARM launcher by Thomas Rast.

3. System Requirements & Installation

3.1. System Requirements

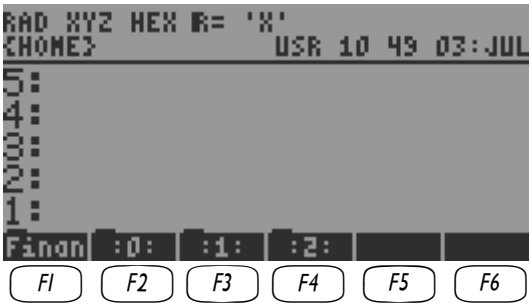
Library 1672: Financial Mathematics has been coded and compiled with Debug2 / Debug4x and the HP49G and is written in System RPL. It has been tested and developed with Beta ROM 1.19-5 and 1.19-6 in RPN-Mode. Using this library requires RPN mode (systemflag 95 set).

3.2. Installation

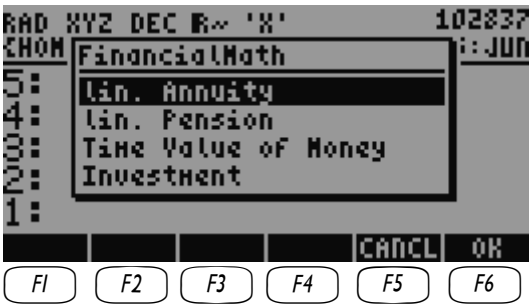
Transfer the program 'FinancialMathematics' to your HP 49G and store it in a port. For speed reasons Port 0 is recommended. After a warmstart (ON and F3) the library will be attached to the {HOME} – directory. Either use the library menu to access the program or ← FINANCE .
Further information about library objects and memory management can be found in the documentation provided with your calculator.

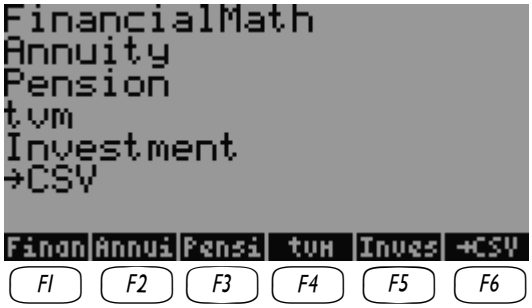
4. What this program computes

Library 1672: Financial Mathematics contains six commands.
Switch to the library-menu with ↱ LIB and NXT, if necessary,



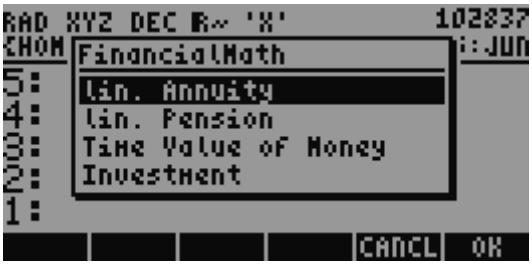
and by pressing the appropriate softkey one changes into the menu of Financial Mathematics.





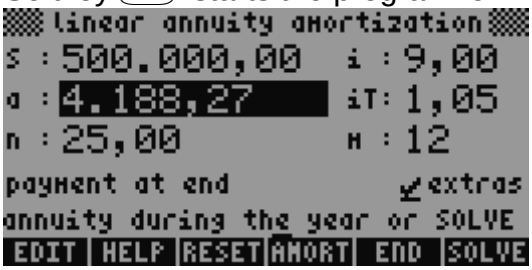
4.1. Commands of the library

Softkey F1 opens a choose-box to allow easily access to the programs of Financial Mathematics.

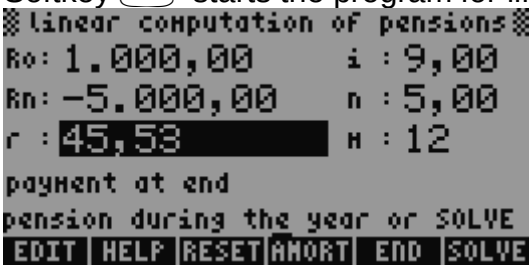


FinancialMath is used as a shortcut and assigned to ↰ FINANCE to replace the built in Financial Solver application.

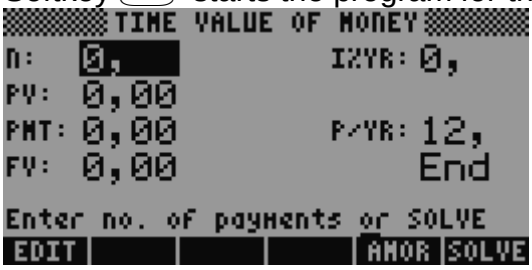
Softkey F2 starts the program for linear annuity amortization.



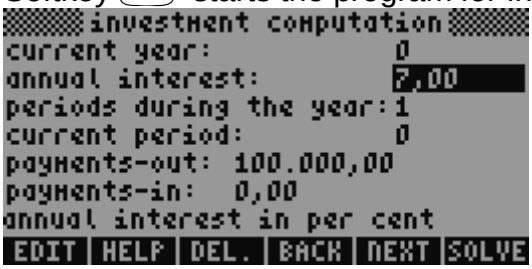
Softkey F3 starts the program for linear computation of pensions.



Softkey F4 starts the program for the built in financial solver.



Softkey F5 starts the program for investment computation.



Softkey F6 converts an existing amortization plan, pension plan or an investment computation into the spreadsheet readable CSV-format.

Please note that the fraction mark in the numbers of the converted file depends on systemflag 51.

4.1.1. Linear annuity amortization

Linear annuity amortization is the process of dividing a constant payment into the amount that applies to interest and the amount that applies to principal until the debt is paid back completely. Payments near the beginning of a loan contribute more interest and less principal than payments near the end of a loan. Using linear interest for the periods during the year – which is the common way used by banks – one has to use the relative interest for the periods i* = i/m. Internally this interest is used by the program.

Details about annuity amortization can be found in various mathematics books.

The following variables are used by this program:

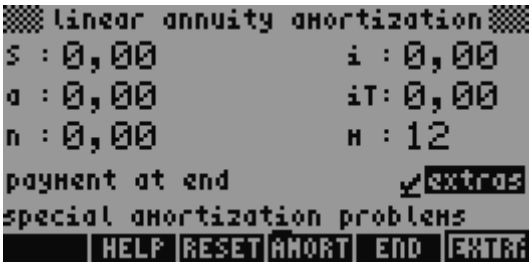
α/δ = agio / disagio (no interest paid on)*
a = annuity during the year
c = number of compounding periods during the year
i = annual interest in per cent
iT = initial annual amortization in per cent
j = annual effective yield (by EU-(AIBD-)method)
g = credit charge (interest paid on)*
m = periods during the year
n = duration in years
RSt = debt due in period t with credit charge but without agio / disagio
S = debt without credit charge and without agio / disagio
t = current period
Tg = accumulated amortization up to t with credit charge and with agio / disagio
Tt = amortization share in period t with credit charge and with agio / disagio
Zg = accumulated interest up to t with credit charge but without agio / disagio
Zt = interest share in period t with credit charge but without agio / disagio
tf/zf = amortization free periods without interest free periods or with interest free periods (negative sign)*

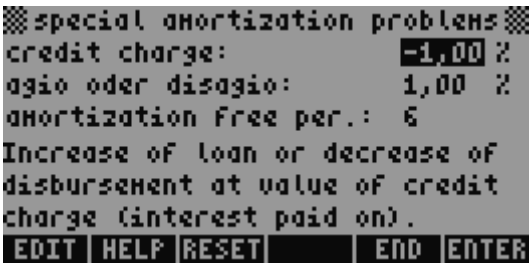
With * marked variables are only searched in the current directory. all other variables will be searched starting from the {HOME} – directory up to the current directory.

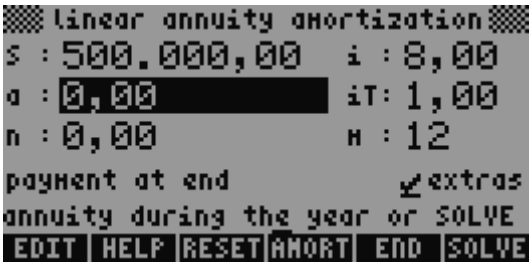
On the main screen “linear annuity amortization” one can solve for the variables S, a, i, iT, and n depending on one other value. If the initial annual amortization in per cent is not known (iT=0) this will automatically be the depending value. If the initial annual amortization in per cent is known but the duration in years (n=0) is unknown this will automatically be the depending value. In all other cases one can choose the dependant value or an error will be generated if not enough information is present.

Example:

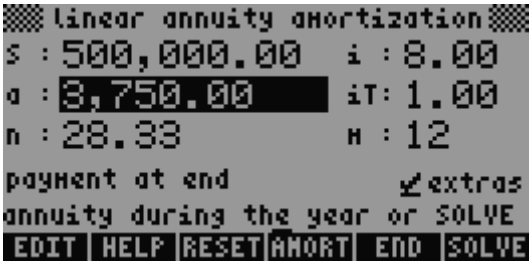
A loan of € 500.000 at 8% annual interest, 1% initial annual amortization, a credit charge of 1% (equal to 99% payment-out) 1% agio and a half year of amortization free periods without interest free periods should be repaid.



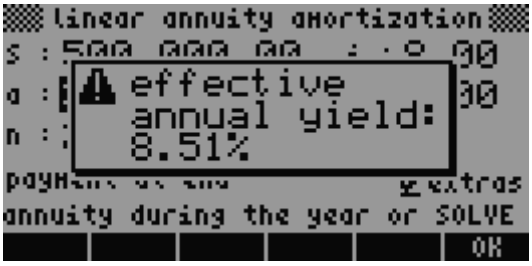




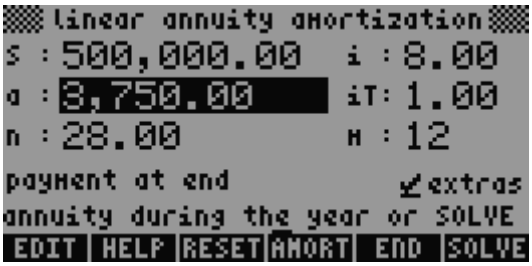
Computing monthly payments and duration in years:

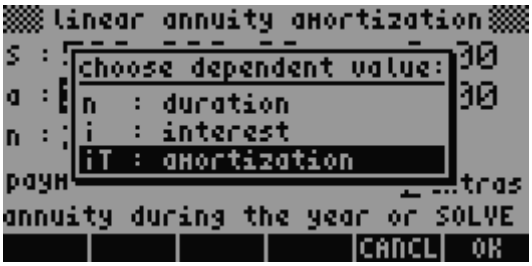


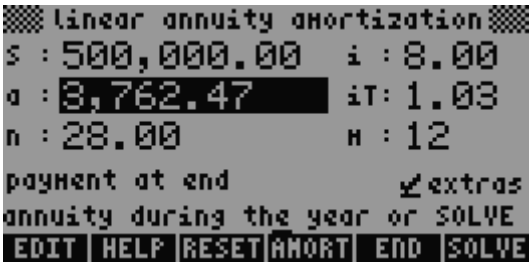
Computing annual effective yield (by EU-(AIBD-)method):



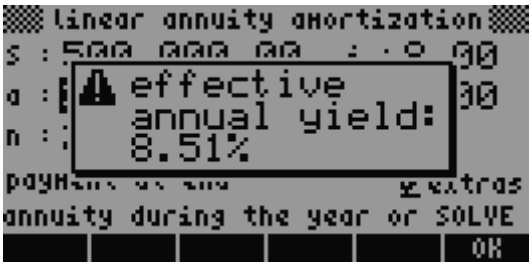
What will be the monthly payment and the initial annual amortization if the duration of the loan should be 28 years ?





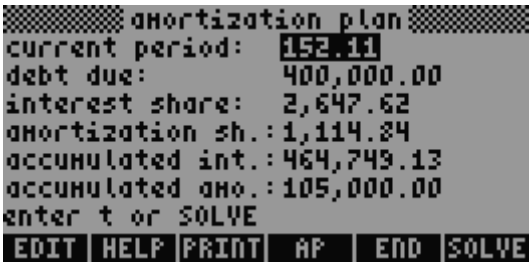


Computing the changed annual effective yield (by EU-(AIBD-)method):

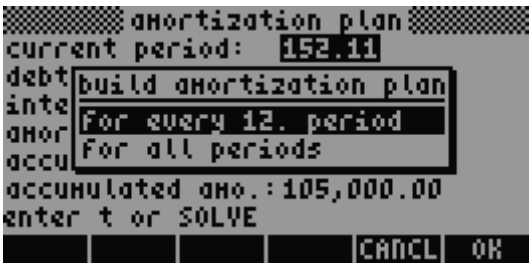


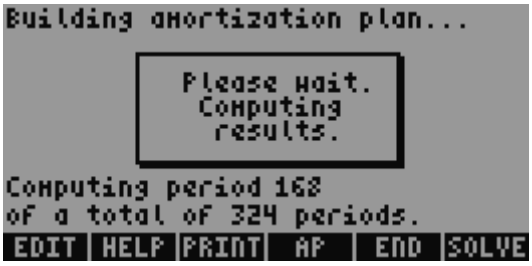
(In this example the annual effective yield changes from 8,5056to 8,5092% but the program shows only two decimal places.)

Next the amortization plan for the loan can be computed. One can either solve for a chosen period or for a chosen debt due. After how many total periods during the year is a debt due of €400.000 ?



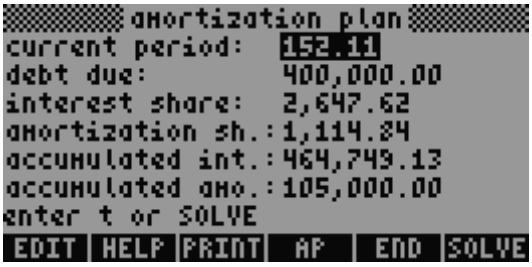
The whole amortization plan or the amortization plan for each last period during the year can be computed by pressing [F2].

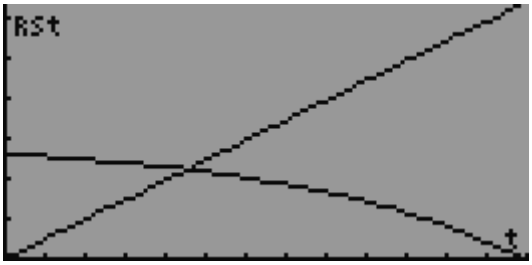




After that the amortization plan can be printed on a serial printer using [F3].

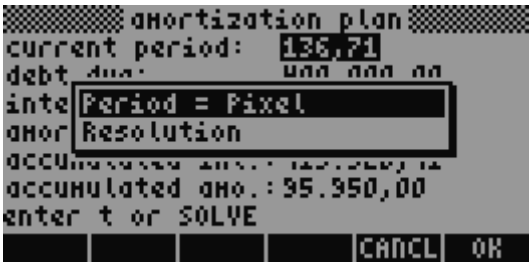
Pressing (NXT) allows to show the amortization plan as a graphic.





[graph] shows the amount of debt due and the payments that are made. The reserved variable EQ will be overwritten but the reserved variable PPAR not. It may be necessary to delete PPAR with [del]. [graph] will create a customized PPAR if this variable is not present.

[zoom] changes the resolution of the graph.



It is possible to choose a resolution so that each pixel represents a period (the total number of periods during a year must be between 130 and 2047 in this case). Resolution defines in which interval the values for the graphic will be computed. The predefined value is every 6. period.

4.1.1.1. Compounding periods different from payment periods

In financial calculations involving a series of payments equally spaced in time with periodic compounding, both periods of time are normally equal and coincident. However, it may happen that the payments periods and the interest periods are different, e.g. Canadian mortgages. In this case it is necessary to replace the nominal annual interest with an adjusted interest rate.

$$adjusted\ interest\ rate = \left(\left(1 + \frac{i}{100 \cdot c} \right)^{\frac{c}{m}} - 1 \right) \cdot 100 \cdot m$$

i = annual interest in per cent
m = number of periods during the year
c = number of compounding periods during the year

4.1.1.2. Used system flags

System flag 13 controls if the help screen for the printer settings will be shown or not if [2nd] [0] is pressed.
System flag 13 set: the help screen for the printer settings will not be shown
System flag 13 clear: the help screen for the printer settings will be shown

System flag 14 controls if payments are made at the beginning or end of a period
System flag 14 set: payment at begin
System flag 14 clear: payment at end

4.1.1.3. Used keys

[EDIT] : Edits the current value of the current field.

[HELP] : Shows the help screen.
[▲] [▶] [ENTER] : next site of the help screen
[◀] [▼] : previous site of the help screen
[ON] : back to the program

[RESET] : Resets all values to the default values. (also [↱] [CLEAR]).

[AMORT] : Switches to “amortization plan”. The values of “linear annuity amortization” will be saved.

[END] : Ends the program or switches back to “linear annuity amortization” Entered values will not be saved.

[SOLVE] : Solves for the current field. Entered values will be saved.

[SPECIAL] : Switches to „special amortization problems“.

[ ] : Computes annual effective yield (by EU-(AIBD-)method). Entered values will be saved.

[PRINT] : Prints the amortization plan.

[ ] : Computes the amortization plan.

[GRAPH] : Shows the amount of debt due and the payments that are made.

[DEL] : Choose box for deleting the graph, PPAR and PDIM.

[RES] : Choose box for changing the resolution of the graph.

[ENTER] : Closes an open command line or ends the program or switches back to “linear annuity amortization”. Entered values will be saved.

[ON] : Stops a running computation or ends the program or switches back to “linear annuity amortization”. Entered values will not be saved.

[NXT] [▲] [▶] [◀] [▼] [⬅] [y^x] [$\sqrt{x}$] [$+/-$] [$1/x$] [÷] [×] [+] [−]
[1] [2] [3] [4] [5] [6] [7] [8] [9] [0] [•] [SPC] using the standard key definition.

4.1.2. Linear computation of pensions

Linear computation of pensions is used for compounded interest calculations that involve regular uniform cash flows called payments. The amount of each payment is the same and occurs at regular intervals. The payment period coincides with the interest compounding period.

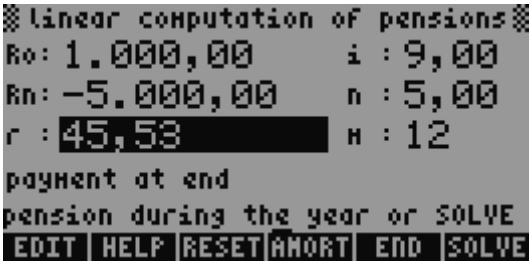
Payments-in are positive, payments-out are negative.

The following variables are used by this program:

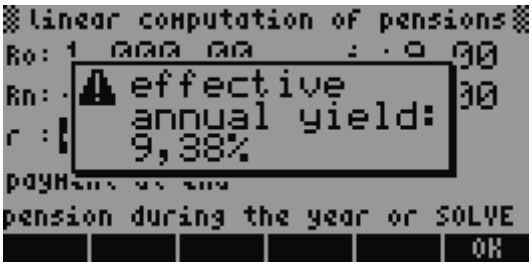
c = number of compounding periods during the year
i = annual interest in per cent
j = annual effective yield (by EU-(AIBD-)method)
m = periods during the year
n = duration in years
r = pension during the year
Ro = present value of the pension (t=0), present value of a total amount of a number of future payments
Rn = future value of the pension (t=n)
RSt = debt due in period t
t = current period
Tg = accumulated amortization up to t
Tt = amortization share in period t
Zg = accumulated interest up to t
Zt = interest share in period t

Example:

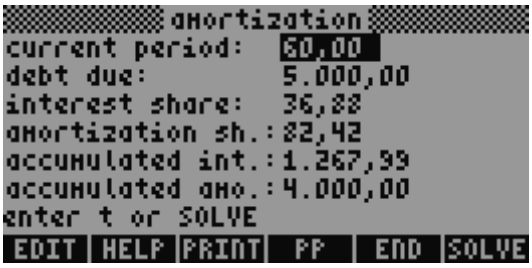
A saving account of € 1000 with 9% annual interest should grow to € 5.000 in 5 years. What is the monthly amount that has to be saved to accomplish this ?



Computing annual effective yield (by EU-(AIBD-)method):

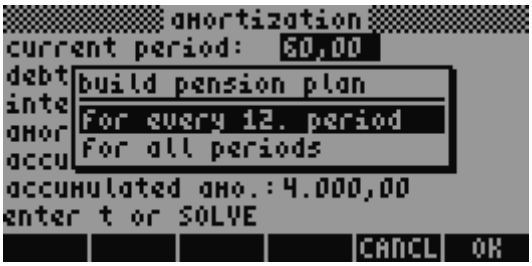


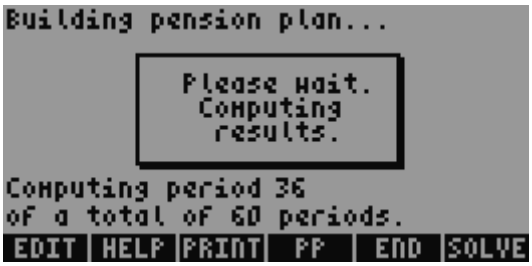
Computing the amortization plan for the saving account:



Amortization calculations determine the amounts applied towards principal and interest in a payment or series of payment. If this is a debt due or a saving depends on the point of view.

The whole pension plan or the pension plan for each last period during the year can be computed by pressing [▓▓12▓▓].





After that the pension plan can be printed on a serial printer using [▓▓PRINT▓▓].

4.1.2.1. Used system flags

System flag 13 controls if the help screen for the printer settings will be shown or not if [▓▓PRINT▓▓] is pressed.
System flag 13 set: the help screen for the printer settings will not be shown
System flag 13 clear: the help screen for the printer settings will be shown

System flag 14 controls if payments are made at the beginning or end of a period
System flag 14 set: payment at begin
System flag 14 clear: payment at end

4.1.2.2. Used keys

EDIT : Edits the current value of the current field.

HELP : Shows the help screen.
▲ ▶ ENTER : next site of the help screen
◀ ▼ : previous site of the help screen
ON : back to the program

RESET : Resets all values to the default values. (also ↱ CLEAR).

AMORT : Switches to “amortization plan”. The values of “linear computation of pensions” will be saved.

END : Ends the program or switches back to “linear computation of pensions” Entered values will not be saved.

SOLVE : Solves for the current field. Entered values will be saved.

i : Computes annual effective yield (by EU-(AIBD-)method). Entered values will be saved.

PRINT : Prints the pension plan.

RUN : Computes the pension plan.

ENTER : Closes an open command line or ends the program or switches back to “linear computation of pensions”. Entered values will be saved.

ON : Stops a running computation or ends the program or switches back to “linear computation of pensions”. Entered values will not be saved.

NXT ▲ ▶ ◀ ▼ ⬅ y^x $\sqrt{x}$ $+/-$ $1/x$ ÷ × + −
1 2 3 4 5 6 7 8 9 0 • SPC using the standard key definition.

4.1.3. Finance Solver

see user guide

4.1.4. Investment computation

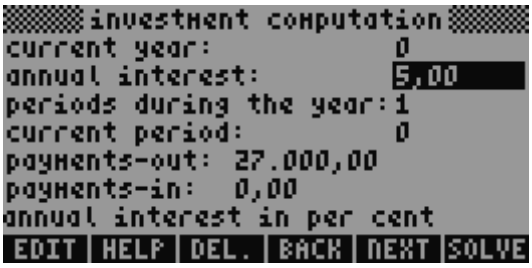
Investment computation is used to compare a financial investment with alternate financial investments.

The following variables are used by this program:

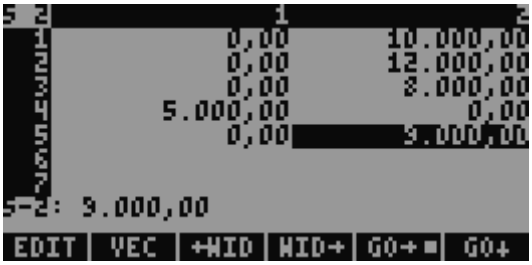
ΣINV= current data matrix
IRR = internal rate of return
MZT = median payment date
NBW= net present value
NER = net uniform series
NEW= net future value
T = time for amortization

Example:

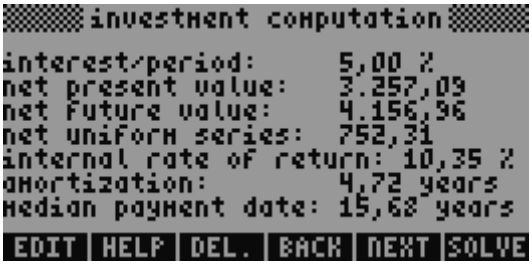
An investment leads to a payments-out of € 27.000. It is expected that this investment will lead to the following results: payments-in of € 10.000 in the first year, payments-in of € 12.000 in the second year, payments-in of € 8.000 in the third year. For the fourth year a a payments-out of € 5.000 is expected and for the fifth year a payments-in of € 9.000.



Entering the data for payments-out and payments-in can either be done in "investment computation" or via [MAT] through the MatrixWriter.



After all data has been entered compute the investment:



Results must be examined critical.

4.1.4.1. Used keys

[EDIT] : Switches to MatrixWriter for entering data for payments-out and payments-in.

[HELP] : Shows the help screen.
[ON] : back to the program

[DEL] : Deletes the current data from the data matrix.

[BACK] : Jumps back one period.

[NEXT] : Jumps forward one period.

[SOLVE] : Computes investment and shows the result.

[GOTO] : Jumps to the selected period.

[NEW] : Deletes an existing data matrix and restarts the program.

[PRINT] : Prints the investment computation.

[END] : Ends the program. Entered values will not be saved.

[ENTER] : Closes an open command line or ends the program. Entered values will be saved.

[ON] : Stops a running computation or ends the program or switches back to "investment computation". Entered values will not be saved.

[NXT] [▲] [▶] [◀] [▼] [←] [y^x] [$\sqrt{x}$] [+/−] [1/x] [÷] [×] [+] [−]
[1] [2] [3] [4] [5] [6] [7] [8] [9] [0] [•] [SPC] using the standard key definition.

5. Version history

12.07.01	Version 1.0	First public version.
20.07.01	Version 1.01	Fixed some small bugs while canceling the amortization plan.
01.08.01	Version 1.1	Content of the variables will only be shown if they contain real numbers. Added pension plan and printing of pension plan Changed variable R0 to Ro.
19.09.01	Version 1.2	Program will be aborted if not enough free memory is available and the previous content of the stack will be restored. To run all programs 14,3 KB free memory is required plus the memory for the used variables.
27.09.01	Version 2.0	Added investment computation.
30.09.01	Version 2.1	Periods during the year are now shown as integers.
30.10.01	Version 2.2	Starting the programs from the choose box is now possible with the right arrow key.
06.11.01	Version 2.3	Added graphic of the amortization plan.
29.01.02	Version 2.31	Fixed bug in field m.
09.02.02	Version 2.32	Fixed accidentally introduced bug with version 2.31 while solving the amortization plan.
17.06.02	Version 2.4	Fixed some small bugs.
21.10.02	Version 2.41	Fixed some small bugs.
10.07.03	Version 2.5	First English version.
19.07.03	Version 2.6	Added adjusting the interest rate if compounding periods and payment periods not match.
08.09.03	Version 2.7	Minor bugs fixed.
01.08.04	Version 2.8	Minor bugs fixed. Adaptation to the HP 49G+. Some time critical program routines now switch to 203 MHz while running and change back to 75 MHz afterward.
31.12.05	Version 2.9	Fixed bugs in the amortization plan. Speed optimization for all computing program routines.
11.09.06	Version 3.0	Saved amortization plans, pension plans or investment computations may be converted into the spreadsheet readable CSV-format. In choose list the right arrow key may be used instead of the enter key. Reprogramming of the pension plan with significant speed gain.
23.10.06	Version 4.0	Fixed wrong results of linear annuity amortization. Speed up amortization plan.
28.10.07	Version 5.0	Added CSV interface for spreadsheet applicationons.
08.02.08	Version 6.0	Optimized CSV interface.
22.11.09	Version 7.0	Adaptation to ROM 2.15.

6. Known bugs

Fixed in Version 1.2	If not enough free memory is available while the program is running the 49G may crash and/or a memory lost might happen. Calculator crashes if the value in the field for the periods during the year and the values in the fields of "special amortization problems" are incorrect.
Fixed in Version 2.31	If field m contains a negative value and m is the value that the program is solving for the calculator crashes after the error message is shown.
Fixed in Version 2.32	Fixed accidentally introduced bug with version 2.31 while solving the amortization plan.
Fixed in Version 2.41	Median payment date is wrong if computing the time for amortization is not possible.
Fixed in Version 2.7	When solving for RSt in the amortization plan RSt was not saved in the current directory if it already exists in a higher directory.
Fixed in Version 2.8	Fixed a bug in computation of pensions. Solving for n and ending the program afterwards leads to a wrong result saved in n. The program routine for amortization and the pension plan computes wrong results on a 49G+ (this is a bug in the OS of the 49G+ in my opinion). This program routine was adjusted for the 49G+.
Fixed in Version 2.9	Fixed a bug in the amortization plan. If an amortization plan for all periods with a credit charge and an agio or disagio in combination with amortization free periods without interest free periods or with interest free periods is calculated the computed results are wrong. The computed results for amortization free periods with interest free periods in combination with an agio or disagio were wrong.
Fixed in Version 3.0	Fixed a bug in computation of pensions. Computing the results for the 0. period leads to a crash.
Fixed in Version 4.0	Results of linear annuity amortization are wrong if a credit charge which decreases the outpayment and/or a disagio were chosen.