Easy Mortgage Math

Mortgage management program lets you think you have more control over your money than the bank does. Contributed by Dick Wink of Peterborough, Ont.

Anyone who has attempted to use a regular annuities program to calculate house mortgages, may well have been disappointed to discover errors in the results. An error of even a few cents in the monthly payment can make a large difference over the time period of a normal mortgage. The normal annuity program assumes that the compounding period and payment period are the same, whereas a mortgage is usually compounded semi-annually, but paid monthly. Another difference that can occur arises because the "professional" computer print-out rounds all dollar and cent figures to two decimal places. The program outlined here, takes account of both factors and produces correctly the total interest, monthly payment and outstanding debt after n payments. It does not, of course, include property tax and insurance which are often included in the monthly payment. After using such a program you may well decide to re-negotiate your mortgage. I did! Tables 1 and 2 show the status for the first four months of a $23,000 mortgage at 10.5% per annum. By just over doubling the monthly payment, the time is reduced to one quarter and there is a saving of $25,000 in interest.

The program was written for a Texas Instrument S R 52 and should easily "fit" into the more recent T.I. models as well as the Hewlett-Packard programmables. It can be split into two parts for calculators with smaller program memories. The formula for the true monthly interest rate (i*) in decimal rather than percentage form is:

$$i^* = (1 + i/c)^{c/12} - 1$$

where i is the annual interest rate in decimal form and c is the number of compoundings per year. The installment payment that the home-owner makes each month (PMT) is given by the equation

$$PMT = P.\, i^*/(1 - (1 + i^*)^{-12t})$$

where P is the initial dollar value of the mortgage, or principal, and t is the number of years over which the mortgage is amortised. The total interest that the home-owner pays over the full period of the mortgage (a truly horrifying sum) is given by:

$$TOT\ INT = PMT \,.\, (12.t) - P$$

These calculations are all performed in the first portion of the program. The second portion uses an iterative technique to generate the lines of the amortisation table so that the outstanding balance after the nth monthly payment can be calculated.

It is quite simple to derive an expression for outstanding balance but the results will disagree with those produced "professionally" because the calculator will perform the computation to a much greater degree of accuracy than the two decimal places appropriate to financial calculations. The iterative technique conforms to standard practice in mortgage computation. If OP(n - 1) denotes the outstanding principal just prior to the nth payment, then the interest payable at this time is:

$$I(n) = OP(n-1) \,.\, i^*$$

The part of the payment which can be applied to reducing the debt (decrement of principal) is:

$$DP(n) = PMT - I(n)$$

The new principal - or debt after the nth payment has been made - is given by:

$$OP(n) = OP(n-1) - DP(n)$$

These three expressions are all calculated in the loop illustrated in the flowchart. Referring to the flowchart, it will be seen that the principal (P), annual interest rate per cent (i%), number of compoundings (c) (Usually c = 2, semi-annual compounding) and time (t) years over which the mortgage is amortised are entered and stored. The number of payments made, (n), is then entered and the program is run. If n =0, only part A is run and the monthly payment and total interest are displayed. If n is non-zero the program enters the loop, initialising the counter and generating succesive lines of the amortisation table until the nth payment is reached and the program halts. Outstanding principal, decrement of principal and interest portion of the payment are in memory and can be displayed. If a printer is available, a very minor modification can effect a printout of the entire amortisation table and not just the nth line. The total program uses 12 memories, 3 levels of parenthesis and one single level subroutine for rounding of the dollars and cents figures.

Table 1. For Principal P=$23000, Rate i%=10.5% compounded semi-annually over 20 years. Monthly payment (PMT) =$226.20

Payment #	Outstanding Principal Before Payment OP(n-1)	Interest I(n)	Reduction in Principal DP(n)	Outstanding Principal After Payment OP(n)
0				23000.00
1	23000.00	196.98	29.22	22970.78
2	22970.78	196.73	29.47	22941.31
3	22941.31	196.48	29.72	22911.59
4	22911.59	196.23	29.97	22881.62

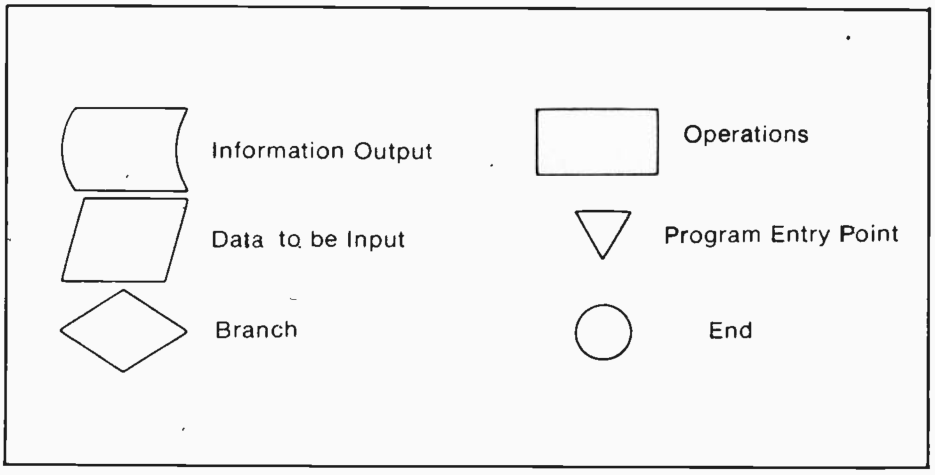
Total interest paid = $31288.00

Table 2. For P=$23000, i%=10.5% compounded semi-annually over 5 years. Monthly payment (PMT) =$491.83

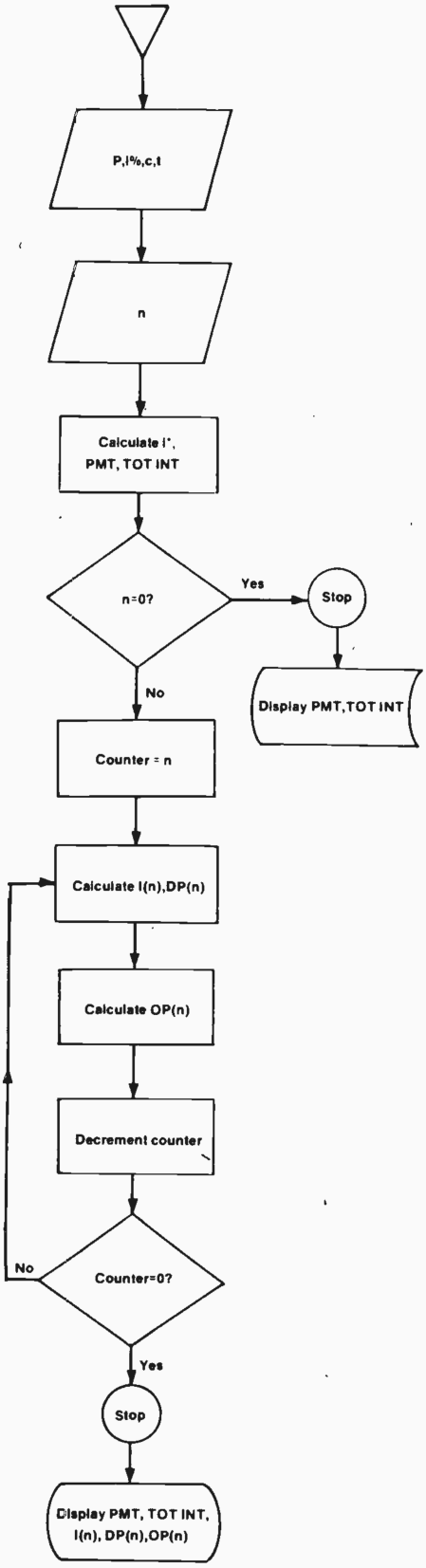
Payment #	Outstanding Principal Before Payment OP(n-1)	Interest I(n)	Reduction in Principal DP(n)	Outstanding Principal After Payment OP(n)
0				23000.00
1	23000.00	196.98	294.85	22705.15
2	22705.15	194.46	297.37	22107.86
3	22107.86	191.91	299.92	21805.37
4	21805.37	189.34	302.49	21502.88

total interest paid = $6509.80

FLOW CHART SYMBOLS



FLOW CHART



ETI Softspot — Easy Mortgage Math

PROGRAM

LOC	CODE	KEY	LOC	CODE	KEY	LOC	CODE	KEY	LOC	CODE	KEY	LOC	CODE	KEY	LOC	CODE	KEY
000	46	LBL	035	09	9	070	75	-	105	02	2	140	01	1	175	81	HLT
	11	A		42	STD		01	1		65	×		00	0		46	LBL
	42	STO		01	1		54	)		43	RCL		65	×		17	B'
	00	0		00	0		42	STO		00	0		43	RCL		43	RCL
	01	1		43	RCL		01	1		04	4		01	1		00	0
005	81	HLT	040	00	0	075	01	1	110	95	=	145	01	1	180	07	7
	46	LBL		01	.1		55	÷		75	-		95	=		81	HLT
	12	B		65	×		53	(		43	RCL		51	SBR		46	LBL
	42	STO		53	(		53	(		00	0		87	1*		18	C'
	00	0		53	(		01	1		01	1		42	STO		43	RCL
010	02	2	045	53	(	080	75	-		95	=	150	00	0		01	1
	81	HLT		01	1		53	(		42	STO		08	8		00	0
	46	LBL		85	+		01	1		00	0		75	-	187	81	HLT
	13	C		43	RCL		85	+		07	7		43	RCL		46	LBL
	42	STO		00	0		43	RCL		43	RCL		00	0		19	D'
015	00	0	050	02	2	085	01	1	120	00	0	155	06	6	190	43	RCL
	03	3		55	÷		01	1		05	5		95	=		00	0
	81	HLT		01	1		54	)		22	INV		42	STO		08	8
	46	LBL		00	0		45	y^x		90	if zro		00	0		81	HLT
	14	D		00	0		53	(		01	1		09	9		46	LBL
020	42	STO	055	55	÷	090	43	RCL	125	03	3	160	44	SUM	195	10	E'
	00	0		43	RCL		00	0		00	0		01	1		43	RCL
	04	4		00	0		04	4		57	FIX		00	0		00	0
	81	HLT		03	3		65	×		02	2		58	dsz		09	9
	46	LBL		54	)		01	1		81	HLT		01	1		81	HLT
025	15	E	060	45	y^x	095	02	2	130	42	STO	165	03	3	200	46	LBL
	42	STO		53	(		94	+/-		00	0		09	9		87	1*
	00	0		43	RCL		95	=		00	0		57	FIX		57	FIX
	05	5		00	0		51	SBR		43	RCL		02	2		02	2
	00	0		03	3		87	1*		00	0		81	HLT		52	EE
030	42	STO	065	55	÷	100	42	STO	135	01	1	170	46	LBL	205	22	INV
	00	0		01	1		00	0		42	STO		16	A'		52	EE
	08	8		02	2		06	6		01	1		43	RCL		22	INV
	42	STO		54	)		65	×		00	0		00	0		57	FIX
	00	0		54	)		01	1		43	RCL		06	6		56	RTN

EXECUTION

Procedure		Enter		Press	Display
1	Enter mortgage details	Principal		A	P
		Rate % P.A.		B	i%
		Compundings/Year		C	i
		Time-Years		D	t
		Number of payments		E	0.00
2	Recover results		2nd	A	PMT
	2nd		2nd	B	TOT INT
			2nd	C	OP (n)
			2nd	D	I (n)
			2nd	E	DP (n)
3	For arbitary payment*, time not specified follow step 1 entering P, i%, c. Enter 30 for t and 0 for n.	Arbitary Payment:		STO 0 6 STO 0 5 GTO 130 RUN	
	Then return to step 2				

*Unusual - but it can happen in some older private mortgages.

SOFTSPOT is ETI's programmable calculator software department. We know there are many of you who have gone to a lot of effort to write routines for your machines — how about sharing the fun. Send us a copy of your pet program, preferably with flow chart. To make things interesting we will restrict our choices to only those programs making use of loops or conditionals.

All programs we publish will be paid for.

Mail to: ETI Softspot
Unit 6, 25 Overlea Blvd.,
TORONTO, Ontario
M4H 1B1

Don't forget to mention what kind of calculator you use — and we'd also be interested to know where you bought it.