

Amortization Schedule Computation

Gary D. Knott, Ph.D.
Civilized Software, Inc.
12109 Heritage Park Circle
Silver Spring, MD 20906 USA
Tel. (301) 962-3711
email: csi@civilized.com
URL: www.civilized.com

One aspect of MLAB is that it is a spreadsheet for the mathematically literate. For example, the MLAB statement sequence to construct a level payment schedule for a loan of a dollars at r percent interest, to be repaid over n months is given below in a do-file called pay.do. For example, to compute the level payment schedule for \$9,000 to be repaid over 16 months at 9.5% annual interest, we type: `do pay`, and enter the values 9000,16, and 9.5 in response to the corresponding prompts. The payment schedule shown below is produced. This do-file can also be used to compute any of either the loan amount, the number of months of the loan, the annual percentage interest rate, or the monthly payment, where the other three values are known.

```
"filename:pay.do = computation of amortization schedule."
reset
echodo=0; namesw=0

type "You will be requested to enter 4 quantities:"
type "a=loan amount,"
type "n=number of months of the loan,"
type "r=annual percentage interest rate, and"
type "q=monthly payment amount."
type "If any one of these input-values is specified to be -1,"
type "then MLAB will compute that value based on the other values."
a=kread("enter loan amount a=");
n=kread("enter number of months n=");
r=kread("enter annual percentage interest rate r=")/1200;
q=kread("enter monthly payment amount q=");

function p(j)=a*(1+r)^j+q*(1-(1+r)^j)/r; "total principal due at month j";
function av()=q*((1+r)^n-1)/(r*(1+r)^n); "loan amount";
```

```

function nv()=log(q/(q-r*a),1+r); "number of months of the loan"
function rv()=root(r,1e-6,100,p(n)); "monthly interest rate";
function qv()=a*(1+r)^n/((1+r)^n-1)*r; "monthly payment amount";

if a<0 then a=av(); /*compute the loan amount*/
if n<0 then n=nv(); /*compute the number of months of the loan*/
if r<0 then r=rv(); /*compute the monthly interest rate*/
if q<0 then q=qv(); /*compute the monthly payment amount*/

type " "
type "the loan amount:"+a
type "the number of months of the loan:"+n
type "the annual percentage interest rate:"+(r*1200)
type "the monthly payment amount:"+q

d=p on 0:(n-1); /*compute the principal due */
i=r*d; /*compute the interest payment*/
nl=1:n
m=nl&'d&'i&'(q-i)&'q
type " "; type "Do you want to print out the payment schedule?"
y=kread("type 1 for YES, 0 for NO:");
if y=1 then \
{type " [prin. due | int. paid | prin. paid | monthly pay.]",m}

type "Total interest paid:"+(n*q-a)
type "Total amount paid:"+(n*q)

draw nl&'d
top title "Total Principal remaining vs Month"
left title "total principal remaining"
bottom title "Month"
yaxis w.yaxis format (-3,6,0,0,2,0)
view
w1=w; blank w1

draw nl&'i lt dashed
draw nl&'(q-i)
top title "interest & principal paid monthly"
left title "dollar amount"
bottom title "Month"

```

```

title "[dashed=interest per month, solid=principal per month]" \
  at (0,1.018) ifract size .14 inches
view

```

Here is an example of running pay.do

```

* do pay
  You will be requested to enter 4 quantities:
  a=loan amount,
  n=number of months of the loan,
  r=annual percentage interest rate, and
  q=monthly payment amount.
  If any one of these input-values is specified to be -1,
  then MLAB will compute that value based on the other values.
  enter loan amount a= 9000
  enter number of months n= 16
  enter annual percentage interest rate r= 9.5
  enter monthly payment amount q= -1

  the loan amount: 9000
  the number of months of the loan: 16
  the annual percentage interest rate: 9.5
  the monthly payment amount: 601.09756

```

Do you want to print out the payment schedule?

```

type 1 for YES, 0 for NO 1
  [prin. due | int. paid | prin. paid | monthly pay.]
1  9000      71.25      529.84756  601.09756
2  8470.15244 67.0553735  534.042186 601.09756
3  7936.11025 62.8275395  538.27002  601.09756
4  7397.84023 58.5662352  542.531324 601.09756
5  6855.30891 54.2711955  546.826364 601.09756
6  6308.48255 49.9421535  551.155406 601.09756
7  5757.32714 45.5788399  555.51872  601.09756
8  5201.80842 41.1809833  559.916576 601.09756
9  4641.89184 36.7483104  564.349249 601.09756
10 4077.5426  32.2805455  568.817014 601.09756
11 3508.72558 27.7774109  573.320149 601.09756
12 2935.40543 23.2386263  577.858933 601.09756
13 2357.5465  18.6639098  582.43365  601.09756

```

```

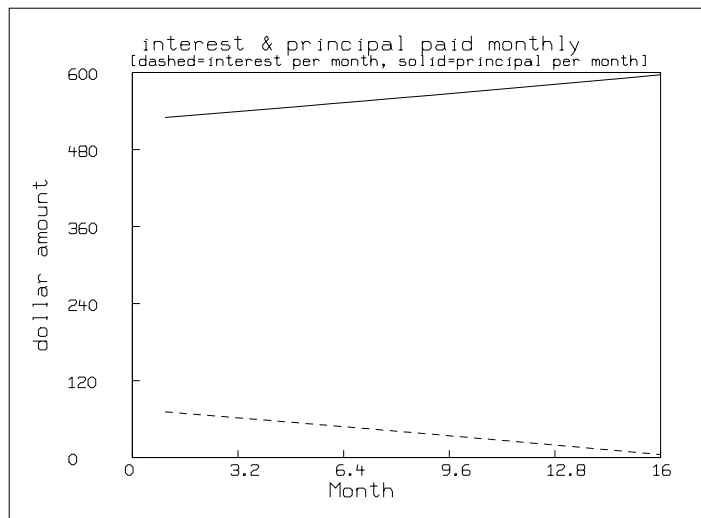
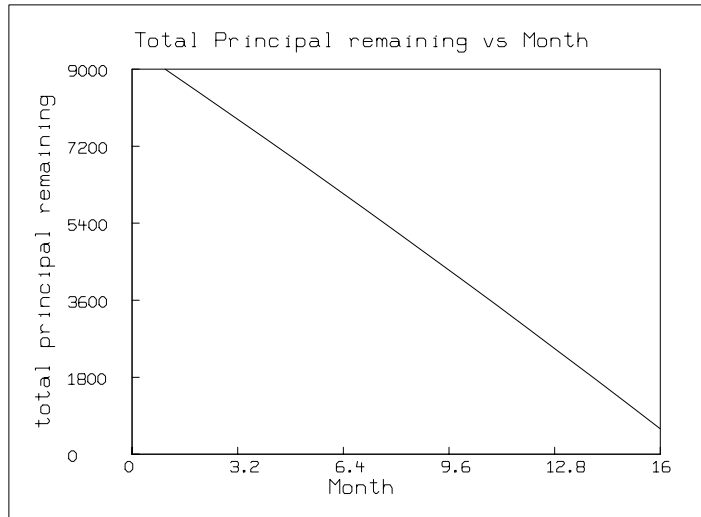
14 1775.11285 14.0529767 587.044583 601.09756
15 1188.06827 9.40554044 591.692019 601.09756
16 596.376248 4.72131196 596.376248 601.09756

```

Total interest paid: 617.560952

Total amount paid: 9617.56095

* exit



Here is another example of running pay.do

* do pay

You will be requested to enter 4 quantities:

a=loan amount,

n=number of months of the loan,

r=annual percentage interest rate, and

q=monthly payment amount.

If any one of these input-values is specified to be -1,
then MLAB will compute that value based on the other values.

enter loan amount a= 200000

enter number of months n= 180

enter annual percentage interest rate r= 9

enter monthly payment amount q= -1

the loan amount: 200000

the number of months of the loan: 180

the annual percentage interest rate: 9

the monthly payment amount: 2028.53317

Do you want to print out the payment schedule?

type 1 for YES, 0 for NO 0

Total interest paid: 165135.97

Total amount paid: 365135.97

* exit

