

DEFine FuNction.....	1
FROM EDITOR'S DESK.....	2
FORMATING CARTRIDGE.....	3
MAKE COFFER.....	5
COMPUTER THERAPY.....	6
LISTER prog.	3
AMORTIZATION prog.	6

The QUANTUM LEPP Newsletter of the Q.L. USERS INTERNATIONAL

Dedicated to Useful Applications Use of the QL



BASIC PROGRAM SERIES:

DEFine FuNction

by TIMOTHY SWENSON
WASHINGTON, D.C.

THE DEFine FuNction command in SuperBasic allows the user to add functions to SuperBasic. Some functions are already built-in, like: COS, SIN, INT, etc. Without the DEFine FuNction command the user can still write his own functions, but with the command the user uses his functions just like built-in functions.

TAKE a look at the sample function listed below. The function is called ROUND. It rounds a number to an integer. If the decimal place is less than .5 then it rounds down. Otherwise it rounds up.

WHEN the user defines a function he has to specify what sort of entry variables are needed. Note that in line 100 that there is an X in the parentheses. This is the variable or number that the function evaluates on. The user may specify more than one variable to evaluate on. The number of variables sent to the function must equal the number of variables that the function evaluates on. If it needs three variables then the user must send it three variables.

THE next lines makes the two variables used by the function LOCAL or temporary. LOCAL variables are used only in the function. Once the function is finished the variables disappear from memory. This way they are not confused with other variables used in other parts of the program, plus it saves memory.

A RETURN command is used to send the results of the function back to the line that called it. There may be more than one RETURN in the function. There must be a RETURN for any case that the function may run accross. In other words the function must always return something.

THE function may be used any part of the program just like any other function.

Define Function cont.:

```

10 LET a = round. (1,3)
20 LET b = round (y)

```

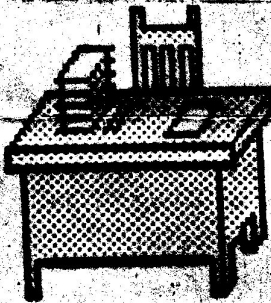
ALL of these are valid commands.

I could go into much more detail about functions but this should give you the general idea of how they are used and how they are created.

```

100 DEFINE FUNCTION round (x)
110   LOCAL temp1, temp2
120   LET temp1 = INT (x)
130   LET temp2 = x - temp1
140   IF temp2 < .5 THEN
150     RETURN temp1
160   ELSE
170     RETURN (temp1 + 1)
180 END DEFINE round

```

**FROM EDITOR'S DESK**

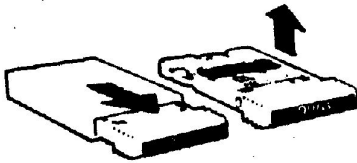
by WILLIAM E. FISCHER

(ED NOTE:) WELCOME to the second issue of the QLUI Newsletter with some programs that are for articles that were written in the first issue. I hope to provide many more programs as each issue comes out. I will need help and ideas for new programs. I am sorry that I am a little going this issue out to you. But I had get the bugs worked out of the programs that are in this issue. An addition problem is the size of the programs has resulted in a couple changes for this issue. So I hope that this will meet with your appraisal. Because of size I will have to wait until the next newsletter to tell you about some new additions the the newsletter. I do think that everyone who are subscribers will really like. So I will be seeing you in the next issue. VF

2

DEFINITION:

function /funk'shan/: a variable so related to another that for for each value assumed by one there is a value determined for the other.



FORMATTING CARTRIDGES:

by BILL WOODWARD
LAKE BULTER, FL.

Even though the first order of business on the QL is the proper use of the microdrive cartridge, the QL User's Guide starts you CLONEing programs without as much as a 'by your leave, sir?' about its USE & ABUSE.

In QLUG's CONCEPT section The DOs & DON'Ts of this little jewel rates three quarters of a page. The capacity of each CARTRIDGE is a minimum of 200 sectors or 100 Kbytes and can store up to 255 sectors.

QDOS uses available free memory space on the cartridge as microdrive buffers, when necessary to improve performance, to store your typing or other data as the case may be. This, I learned the hard way..

On New Years Eve 1986, while typing on a term paper for my daughter until the wee hours of the new year. I had noticed mdv2_ frequently running and shutting off as I typed and realized QDOS had opened a temporary file on mdv2_.

I had no idea how to retrieve the data QDOS had placed on the md. Do you? I searched the QLUG without avail. Finally on a new cartridge I saved what I could, but lost about two pages of typing. An unexpanded QL will store about two and a half pages in its memory before it fills and QDOS starts a temporary file on md2_.

A new cartridge needs FORMATTing several times in order to stretch the tape and condition the cartridge. Since FORMATTing several cartridges takes time. Here is a small program that will FORMAT as many cartridges as you wish and give you time to do more necessary chores. It consists of two-ten pass loops in tandem nested in a third loop that determines how many times you will need these ten pass loops.

```
100 CLS
110 PRINT\ "Enter number of cassettes to be FORMATTed"
120 INPUT\ "Number of cassettes";n
130 FOR x=1 TO n
140 for a=1 TO 10
150 PRINT a;
160 FORMAT mdv1_
```

Formatting Cart. cont.:

```

170 END FOR a
180 FOR b=1 TO 10
190 PRINT b;
200 FORMAT mdv2_
210 END FOR b
220 CLS
230 CSIZE 3,1:AT 4,10:PRINT 'HOORAY!':CSIZE 0,0
240 PAUSE 200
250 END FOR x

```

Load cartridges to be FORMATTed in mdv1_ and mdv2_ and run the program. In line 120 INPUT 1/2 the number of cartridges you will FORMAT. This sets the x-loop x=1 TO n(1/2 the number of cartridges to be formatted). Mdv1_ runs and FORMATs ten times printing the event number and the number of sectors each time. Replace FORMATTed cartridge in mdv1_ when mdv2_ runs and FORMATs ten times printing the event number and number of sectors. When both mdv1_ & mdv2_ have completed their loop the screen will clear and 'HOORAY' in large letters will be printed in the middle of your screen. Replace the cartridge in mdv2_.

In the QLUI, it is the sole responsibility of the contributing authors to insure that no copyright infringements have occur in their respective contributions. The QLUI Newsletter and the Editor/Publisher is not are responsible for any infringements. Original material printed in this Newsletter may only be reprinted by other T/S user groups and educational institutions without prior written consent by either the Editor/Publisher or the authors. All others must first obtain written permission prior to reprint. An acknowledgement of the QLUI and the author must be included. NOTE: Opinions expressed in this Newsletter are those of the author of the particular article and are not necessarily those of the QLUI or the Editor/Publisher.

MAKE COFFEE cont.:

about the program you are going to check. You will have to answer them before going on. The first is the title of the program, next is the starting point which can be one or if you know the exact starting number. The next is the finishing number which you can use 99999 or what ever is the the last number used + 1. And finally the question 'Do you want to have a print out?' I hope that you find this program a great help and use it as much as possible. Each program will be have the checksum with it.

I will get into more concerning the checksum program and how to ensure the correcting of the basic programs you have entered into your computer.



MAKE COFFEE

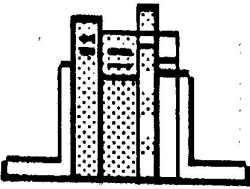
USING CHECKSUM PROGRAM

by WILLIAM FISCHER
ORANGE PARK, FL.

HOW would you look at yourself, as an expert on computers or just average in dealing with a computer? I would say that just about everyone has reach a point of fristration. This is in dealing with the entering of a Basic program from a book or magazine to a computer. This has made us all wonder if it was correct. This is even more so with a longer programs. Just think about this for a minute and raise your right hand if it is true. Now there are those who are able to know that they have enter all lines of a program correctly. To those individuals I want to give them a big hand. However I know that there is a greater number of us that are not completely sure, that they have correctly entered the program. I have to put myself in this group. This main group are the ones that I am directing this article toward. This is not to say that I am leaving the others out. In fact those individuals that can enter programs correctly can be of a great help to us by giving us any ideas or suggestions that will make easier for us. This is as long as they are willing to pass it on. Most individuals have questions while entering a basic program into their computers.

WELL for those who have not heard or are not completely familiar with one way of ensuring that their programs are correctly entered properly. There is a checksum generated program. This program provides a generated number for each line of a basic program entered into a computer. I know that many of you are saying 'what is a checksum!'. It is a program that reads each line of a basic program and then counts the number of bytes and then prints them out for you. I am still working on just how it completly works, myself. The program that I have at present and am using, was obtained for a QL book called 'THE WORKING SINCLAIR QL' by David Lawrence. It provides a listing of numbers for each line. However I am starting to work on making some changes to this program to match some of the other checksum programs that are available for the other computers and for software basic programs. This program is a good one and I will use it and provide the checksum numbers for all programs that are printed in the QLUI newsletter. So everyone will have this numbers available to them as they put their copy of the program into their computers.

NOW I will explain how this program will provide you with the numbers. You first load the 'CSG Program' into the computer with `lrun device_csg`. Now there will be a couple of questions



COMPUTER THERAPY

by BILL WOODWARD
LAKE BUTLER, FL.

(ED NOTE:) THIS article is the second part of the article written about the QL computer written by Bill Woodward and shows how the computer can be a therapy tool. WF.

THERE IS NO SUCH THING AS A FREE MEAL or a computer that runs itself as you wish it to do. I know what you are experiencing when you sit down to the computer and stare the keyboard and the cursor blinks back at you.

I am going to try to remember the times I have wondered to myself, "Which key do I punch to get this thing started?". Then I will write out an article about WHAT DO I DO AFTER I TURN ON THE SWITCH.

I have had to buy software, firmware, and hardware to get my computer to obey me. I am still in the novice class after eight years. I am not afraid to ask questions, or to have any one to show me how to do something. Most of what work with was developed or inspired by others. Two operations that are a must in computing is to SAVE and BACKUP frequently. 1) DO NOT LEAVE YOUR COMPUTER EVER, WITHOUT SAVING WHAT YOU WORKING ON! 2) DO NOT QUIT FOR THE DAY WITHOUT BACKING UP YOUR DISK OR MIRCO CARTRIDGE. When you have worked for hours and failed to SAVE or BACKUP and had sudden lighting strike and your monitor looks like a finger painting by Gargantua, you will wish you had both SAVED and BACKUPped! Believe me.

SO I put my Bank Statement Reconiliation ideas to work on the QL spread sheet. You saw the results on the front page of the QLusa, vol. 1988/issue #1. A normal programmer should write a similar programer in a couple of hours, but it took me several years. Each idea had to be developed and tested to see if it did the work that I needed. Some ideas came to me in the middle of a deep sleep and by the time I was awake, they had vanished but returned at a later date. Since my strokes, I have had to remove the word FRUSTRATION from my vocabulary and replaced it the word INSPIRATION, Now when something will not do the work, instead of getting hot and blowing up, I remain calm and use the cause of my failure as an incentive to correct the problem. I was warned to stay out of all stressful situations.

6

NEVER use two words when you can find one that will do just as good.

File Name = csg

```
1 1000 REMark *****
2 1010 REMark control
3 1020 REMark *****
4 1030   INK 7 : PAPER 3 : CLS
5 1040   count=0
6 1050   INPUT "Name of Program: ";program$
7 1060   INPUT "Start Line: ";start
8 1070   INPUT "Finish Line: ";finish
9 1080   program$="mdvl_" & program$
10 1090   INPUT "Printer output (Y/N): ";hard$
11 1100   hard=hard$="Y" OR hard$="y"
12 1110   IF hard
13 1120       OPEN #7, ser1
14 1130   ELSE
15 1140       OPEN #7,scr
16 1150       INK#7,7 : PAPER#7,3
17 1160       CLS#7
18 1170   END IF
19 1180   PRINT#7,"CHECKSUM TABLE FOR!..";program$
20 1190   PRINT#7,"Start line is!..";start
21 1200   PRINT#7,"Finish line is!..";finish\
22 1210   OPEN #8,program$
23 1220   :
24 1230   :
25 1240   REPEAT lines
26 1250       INPUT #8,line$
27 1260       line_number=line$
28 1270       IF line_number>finish THEN EXIT lines
29 1280       IF line_number>=start
30 1290           analyse
31 1300           check_print
32 1310       END IF
33 1320       IF EOF(#8) THEN EXIT lines
34 1330   END REPEAT lines
35 1340   :
36 1350   :
37 1360   PRINT#7 : CLOSE #7
38 1370   CLOSE #8
39 2000 REMark *****
40 2010 DEFINE PROCEDURE check_print
41 2020 REMark *****
42 2030   IF line_number/1000=INT(line_number/1000)
43 2040       PRINT#7,\\line_number;          "*****"
44 2050       count=0
45 2060   END IF
46 2070   number$="          "& checksum
47 2080   number$=number$(LEN(number$)-4 TO)
48 2090   PRINT#7,!number$;
49 2100   count=count+1
50 2110   IF hard AND count=6
51 2120       count=0
52 2130       PRINT#7
53 2140   END IF
54 2150 END DEFINE check_print
55 3000 REMark *****
```

File Name = csg

```
56 3010 DEFine PROCedure analyse
57 3020 REMark *****
58 3030     checksum=0
59 3040     FOR i=1 TO LEN(line$)
60 3050         IF line$(i)<>" "
61 3060             IF i/2=INT(i/2)
62 3070                 checksum=checksum+CODE(line$(i))
63 3080             ELSE
64 3090                 checksum=checksum-CODE(line$(i))
65 3100             END IF
66 3110         END IF
67 3120     NEXT i
68 3130 END DEFine analyse
```

File Size = 1863
Total Pages = 2
Total Lines = 68

CHECKSUM TABLE FOR:..mdvl_csg

Start line is:..1

Finish line is:..9999

1000*****

-43	-95	-45	470	93	151
-168	53	119	-212	53	-19
-115	4	92	289	88	55
246	205	141	63	57	56
-157	45	98	90	-21	75
120	63	-197	-96	56	55
187	30				

2000*****

-44	-263	-46	17	104	91
54	-8	-79	107	166	139
95	93	57	-92		

3000*****

-45	-229	-47	4	112	-143
-159	-78	-4	-82	60	59
-96	-56				

```

1 10 WINDOW 512,256,0,0:CLS:PRINT"Amortization table program"
2 20 PRINT
3 30 PRINT"Will calculate monthly payment needed to amortize a mortgage"
4 40 PRINT"and then print a complete amortization table given..."
5 50 PRINT"interest rate, principal amount of the loan, starting date &"
6 60 PRINT"either the duration of the loan (in years or months) or"
7 70 PRINT"the monthly payment. The borrowers name can also be included."
8 80 PRINT
9 90 PRINT
10 100 y=0:m=0:yp=0:yi=0:i2=0:mp=0:p2=0:l=1:pg=1:ye=0:fg=0:ef=0
11 110 INPUT"Name ";na$
12 120 IF LEN(na$)>23 THEN PRINT na$(1 TO 23):PRINT"Maximum name length is 23 c
haracters is this alright? (Y/N)": ELSE GO TO 160
13 130 a$=INKEY$:IF a$="" THEN GO TO 130
14 140 IF a$="y" OR a$="Y" THEN GO TO 160
15 150 GO TO 110
16 160 INPUT"Enter principal (Amount Borrowed) ";p1$
17 170 FOR x=1 TO LEN(p1$)-1:IF p1$(x)=", " THEN p1$=p1$(1 TO x-1) & p1$(x+1 TO)
18 180 NEXT x
19 190 INPUT"Enter annual interest rate (8 1/2 % = 8.5) ";r:r=r/100
20 200 m2=r/12
21 210 INPUT"Enter monthly payment or 0 to calculate ";mp:IF mp>0 THEN GO TO 33
0
22 220 np=1
23 230 INPUT"Enter number of years, or 0 if you want to enter months ";y
24 240 IF y>0 THEN m=y*12:GO TO 260
25 250 INPUT"Enter number of months ";m
26 260 mp=p1$*(m2*(1+m2)^m)/(((1+m2)^m)-1):mp=INT(mp*100+.5)/100
27 270 IF (p2-(100*p2-INT(100*p2))<.5 THEN p2=INT(100*p2)/100:GO TO 290
28 280 p2=(INT(100*p2)+1)/100
29 290 PRINT"Monthly payment will be ";mp
30 300 PRINT"Recalculate monthly payment (Y/N)"
31 310 an$=INKEY$:IF an$="" THEN GO TO 310
32 320 IF an$="y" OR an$="Y" THEN AT 10,0:GO TO 160
33 322 INPUT"(P)rinter, (D)isplay, or (B)oth ";pd$:IF pd$="p" OR pd$="P" THEN p=1
34 325 IF pd$="d" OR pd$="D" THEN p=0: ELSE IF pd$="b" OR pd$="B" THEN p=2
35 330 PRINT:PRINT"Beginning date (July 15, 1988 is 7,15,1988)"
36 340 INPUT"Month ";m1:INPUT"Day ";d:INPUT"Year ";y
37 360 a=1:l=1
38 370 CLS:IF p=0 OR p=2 THEN PRINT"Amortization Table ";:IF np=1 THEN PRINT m;
39 375 IF p>0 THEN BAUD 600:CLOSE:OPEN #4, ser2:PRINT #4,"Amortization Table ";
:IF np=1 THEN PRINT #4,m;
40 380 IF p=0 OR p=2 THEN np=0:PRINT " Payments of ";
41 385 IF p>0 THEN np=0:PRINT #4," Payments of ";
42 390 IF p=0 OR p=2 THEN PRINT mp;
43 395 IF p>0 THEN PRINT #4,mp;
44 400 IF p=0 OR p=2 THEN PRINT" at a rate of ";r*100:AT 0,62:PRINT"Page ";pg:l
=1+l
45 405 IF p>0 THEN PRINT #4," at a rate of ";r*100:PRINT #4, TO 62;"Page ";pg:
IF p=1 THEN l=1+l
46 410 IF p=0 OR p=2 THEN PRINT"Pay"TO 5;"Due"TO 11;"Beginning"TO 22;"Interest"
TO 32;"Principal"TO 43;"Ending"TO 51;"Interest Principal":l=1+l
47 415 IF p>0 THEN PRINT #4,"Pay"TO 5;"Due"TO 11;"Beginning"TO 22;"Interest"TO
32;"Principal"TO 43;"Ending"TO 51;"Interest Principal":IF p=1 THEN l=1+l

```

```

48 420 IF p=0 OR p=2 THEN PRINT TO 1;"#"TO 4;"Date"TO 13;"Balance"TO 23;"Paymen
t"TO 34;"Payment"TO 42;"Balance"TO 52;"To Date    To Date":l=l+1
49 425 IF p>0 THEN PRINT #4, TO 1;"#"TO 4;"Date"TO 13;"Balance"TO 23;"Payment"TO
0 34;"Payment"TO 42;"Balance"TO 52;"To Date    To Date":IF p=1 THEN l=l+1
50 430 IF ye=0 THEN IF p=0 OR p=2 THEN PRINT TO 4;"(";y;")":l=l+1;ye=1
51 435 IF ye=0 THEN IF p>0 THEN PRINT #4, TO 4;"(";y;")":l=l+1;ye=1
52 440 IF fg=1 THEN GO TO 1110
53 450 i=INT((p1$m2)*100+.5)/100
54 460 yi=yi+i
55 470 i2=i2+i
56 480 pq=mp-i
57 490 yp=yp+pq
58 500 IF p1$<mp THEN p2=p2+p1$:GO TO 520
59 510 p2=p2+pq
60 520 n=p1$-pq
61 530 ON a GO TO 540,580
62 540 IF pq>0 THEN GO TO 570
63 550 PRINT"Payment set too low"
64 560 GO TO 160
65 570 a=2
66 580 IF n>5E-3 THEN GO TO 620
67 590 yp=yp-pq+p1$
68 600 pq=p1$
69 610 n=0
70 620 IF p=0 OR p=2 THEN PRINT lc;;lc=lc+1
71 625 IF p>0 THEN PRINT #4, lc;;IF p=1 THEN lc=lc+1
72 630 IF p=0 OR p=2 THEN PRINT TO 4;m1;
73 635 IF p>0 THEN PRINT #4, TO 4;m1;
74 640 IF p=0 OR p=2 THEN PRINT"/";d;
75 645 IF p>0 THEN PRINT #4,"/";d;
76 650 IF p=0 OR p=2 THEN PRINT TO 11;INT(p1$*100+.5)/100;
77 655 IF p>0 THEN PRINT #4, TO 11;INT(p1$*100+.5)/100;
78 660 IF p=0 OR p=2 THEN PRINT TO 23;i;
79 665 IF p>0 THEN PRINT #4, TO 23;i;
80 670 IF p=0 OR p=2 THEN PRINT TO 33;INT(pq*100+.5)/100;
81 675 IF p>0 THEN PRINT #4, TO 33;INT(pq*100+.5)/100;
82 680 IF p=0 OR p=2 THEN PRINT TO 43;INT(n*100+.5)/100;
83 685 IF p>0 THEN PRINT #4, TO 43;INT(n*100+.5)/100;
84 690 IF p=0 OR p=2 THEN PRINT TO 52;INT(i2*100+.5)/100;
85 695 IF p>0 THEN PRINT #4, TO 52;INT(i2*100+.5)/100;
86 700 IF p=0 OR p=2 THEN PRINT TO 61;INT(p2*100+.5)/100
87 705 IF p>0 THEN PRINT #4, TO 61;INT(p2*100+.5)/100
88 710 GO SUB 930
89 720 IF n<5E-3 THEN GO TO 850
90 730 IF m1=12 THEN GO TO 760
91 740 m1=m1+1
92 750 GO TO 930
93 760 IF p=0 OR p=2 THEN PRINT"Yearend ";y;" Totals...";
94 765 IF p>0 THEN PRINT #4,"Yearend ";y;" Totals...";
95 770 IF p=0 OR p=2 THEN PRINT TO 22;yi;
96 775 IF p>0 THEN PRINT #4, TO 22;yi;
97 780 IF p=0 OR p=2 THEN PRINT TO 32;yp;GO SUB 930
98 785 IF p>0 THEN PRINT #4, TO 32;yp;GO SUB 930
99 790 yi=0;yp=0
100 800 IF ef=1 THEN GO TO 870
101 810 m1=1

```

File Name = amort

```
102 820 y=y+1:IF p=0 OR p=2 THEN PRINT TO 4;"(";y;")":GO SUB 930
103 825 IF p>0 THEN PRINT #4, TO 4;"(";y;")":GO SUB 930
104 830 pl$=n
105 840 GO TO 450
106 850 BEEP 1,1
107 860 IF ml<>1 THEN ef=1:GO TO 760
108 870 IF p=0 OR p=2 THEN PRINT" Table produced for ";na$;" by Sinclair QL"
109 875 IF p>0 THEN PRINT #4," Table produced for ";na$;" by Sinclair QL"
110 880 PRINT
111 890 PRINT"Another Amortization (Y/N)"
112 900 in$=INKEY$:IF in$="" THEN GO TO 900
113 910 IF in$="Y" OR in$="y" THEN RUN
114 920 STOP
115 930 IF l=19 AND n>5E-3 THEN PRINT"Press any key to continue":l=l+1
116 940 IF l=20 THEN GO TO 970
117 950 l=l+1
118 960 RETURN
119 970 l=1:pg=pg+1
120 980 bn$=INKEY$:IF bn$="" THEN GO TO 980
121 990 fl=l
122 995 tl$=n
123 1000 GO TO 370
124 1110 RETURN
```

File Size = 5266

Total Pages = 3

Total Lines = 124

CHECKSUM TABLE FOR:..mdv1_amort

Start line is:..1

Finish line is:..9999

152	75	-211	-274	-220	-74
405	69	68	-237	-53	-49
93	300	-79	-440	435	66
-135	-104	-536	-58	-142	208
-52	-234	-106	-237	-284	-347
-49	-246	-90	-83	1	-180
-93	-21	-144	-49	98	80
3	219	129	137	-84	152
-97	35	-144	-55	-13	-169
-168	-167	-60	74	-69	-153
83	-9	-521	-77	-132	14
-3	-15	-150	91	-40	-92
45	13	-58	42	-85	11
-52	62	-101	55	-92	117
-152	53	-108	72	7	34
-118	-90	172	-215	39	-80
157	-196	-111	-15	-127	21
-203	-201	-86	17	24	-40
-3	-125	-231	-147	42	-53
-316	27	-213	-35	-155	-51
-54	-205				

1000*****

File Name = lister

```
1 100 REMark ** LISTER by Timothy Swenson
2 110 CLS : INK 0: PRINT \\
3 120 PRINT "      P R O G R A M " L I S T E R"
4 130 PRINT "      by Timothy Swenson"\\
5 140 PRINT "Enter Name of File to List : "
6 150 PRINT " (Enter Drive Name, ie. MDV1_)"
7 160 INPUT infile$
8 170 OPEN_IN #4,infile$
9 180 PRINT "\\ " Enter Print Option "
10 190 PRINT " -----"
11 200 PRINT "      1) Regular Print"
12 210 PRINT "      2) Bold Face"
13 220 PRINT "      3) Elite"
14 230 PRINT "      4) Italic"
15 240 PRINT "      5) Near Letter Quality"\\
16 250 INPUT "Enter choice : ";p_opt
17 260 PRINT
18 270 INPUT "Line Numbers? (y/n) : ";a$
19 280 l_opt=0
20 290 IF a$="y" OR a$="Y" THEN l_opt=1
21 300 CLS
22 310 STRIP 4
23 320 AT 10,10: PRINT " P R I N T I N G "
24 330 STRIP 2
25 340 OPEN #3,ser1
26 350 PRINT #3,CHR$(27);"@";
27 360 PRINT #3,CHR$(27);
28 370 IF p_opt = 2 THEN PRINT #3,"E";
29 380 IF p_opt = 3 THEN PRINT #3,"G";
30 390 IF p_opt = 4 THEN PRINT #3,"M";
31 400 IF p_opt = 5 THEN PRINT #3,"x1";
32 410 LINenum=1: page=0: count=1: size=0
33 420 header
34 430 REPEAT loop
35 440 INPUT #4,in$
36 460 IF l_opt=1 THEN
37 470 IF LINenum<=9 THEN PRINT #3," ";LINenum;" ";in$
38 480 IF LINenum>9 AND LINenum<=99 THEN PRINT #3," ";LINenum;" ";in$
39 490 IF LINenum>99 THEN PRINT #3,LINenum;" ";in$
40 495 IF EOF(#4) THEN EXIT loop
41 500 END IF
42 510 IF l_opt=0 THEN PRINT #3,in$
43 520 LINenum=LINenum+1: count=count+1
44 530 IF l_opt=0 AND LEN(in$)>79 THEN count=count+1
45 540 IF l_opt=1 AND LEN(in$)>75 THEN count=count+1
46 550 size=size+LEN(in$)+1
47 560 IF count=57 THEN footer: header
48 570 END REPEAT loop
49 580 PRINT #3
50 590 PRINT #3,"      File Size = ";size
51 600 PRINT #3,"      Total Pages = ";page
52 610 IF l_opt=1 THEN PRINT #3,"      Total Lines = ";LINenum
53 620 PRINT #3,CHR$(12);
54 630 CLOSE #3
55 640 CLOSE #4
56 650 REMark *****
```

File Name = lister

```
57 660 DEFINE PROCEDURE header
58 670 PRINT #3
59 680 PRINT #3,"      File Name =";infile$(6:10)
60 690 PRINT #3: PRINT #3
61 700 page=page+1: count=1
62 710 END DEFINE
63 720 REMARK *****
64 730 DEFINE PROCEDURE footer
65 740 PRINT #3
66 750 PRINT #3,"      PAGE "page
67 760 PRINT #3,CHR$(12);
68 770 END DEFINE
```

File Size = 2052

Total Pages = 2

Total Lines = 68

CHECKSUM TABLE FOR:..mdvl_lister

Start line is:..1

Finish line is:..9999

-240	-163	811	-35	-175	-462
-152	-124	-42	-185	-34	-211
-152	-183	-142	-231	-121	-191
-157	390	-125	-190	430	-186
-88	8	-46	-122	-120	-114
-181	-10	-37	-26	-118	70
148	234	13	22	9	-11
1	52	50	-95	13	-82
-106	-155	-378	337	-57	-91
-89	-7	81	-108	-231	-225
35	-86	-11	63	-112	-146
-54	-80				



COMING IN NOVEMBER

WELCOME to the second issue of QLUI and you enjoying all of the articles and programs. The following are some of the articles and items that I am planning to have appear in the next issue of the QLUI Newsletter. So I hope that all subscribers and readers will be looking forward to seeing them roll off the press about NOV. 29, 1988.

Basic program series

Subscribers' Public Domain Library

First Impression of the QL

MAKE COFFEE dealing more with the use of Checksum Generated No.

! QLUI Newsletter !
! 419 Parkwood Dr. E. !
! Orange Park, Fl. 32073 !

VOL. 1988/ISS. #2
