



410	DATA	11 FRODO BAGGINS	100
420	DATA	41 BILBO BAGGINS	100
430	DATA	42 ARWEN UNDELING	100
440	DATA	43 EOWEN BRAND	100
450	DATA	44 GALADRIEL	100
460	DATA	45 GANDALF	100
470	DATA	46 GOLLUM	100
480	DATA	47 HALDATH	100
490	DATA	48 HIRION	100
500	DATA	49 ISILDUR	100
510	DATA	50 KHALDUL	100
520	DATA	51 LANTHERNE	100
530	DATA	52 LEBEN	100
540	DATA	53 LITH	100
550	DATA	54 LITH	100
560	DATA	55 LITH	100
570	DATA	56 LITH	100
580	DATA	57 LITH	100
590	DATA	58 LITH	100
600	DATA	59 LITH	100
610	DATA	60 LITH	100
620	DATA	61 LITH	100
630	DATA	62 LITH	100
640	DATA	63 LITH	100
650	DATA	64 LITH	100
660	DATA	65 LITH	100
670	DATA	66 LITH	100
680	DATA	67 LITH	100
690	DATA	68 LITH	100
700	DATA	69 LITH	100
710	DATA	70 LITH	100
720	DATA	71 LITH	100
730	DATA	72 LITH	100
740	DATA	73 LITH	100
750	DATA	74 LITH	100
760	DATA	75 LITH	100
770	DATA	76 LITH	100
780	DATA	77 LITH	100
790	DATA	78 LITH	100
800	DATA	79 LITH	100
810	DATA	80 LITH	100
820	DATA	81 LITH	100
830	DATA	82 LITH	100
840	DATA	83 LITH	100
850	DATA	84 LITH	100
860	DATA	85 LITH	100
870	DATA	86 LITH	100
880	DATA	87 LITH	100
890	DATA	88 LITH	100
900	DATA	89 LITH	100
910	DATA	90 LITH	100
920	DATA	91 LITH	100
930	DATA	92 LITH	100
940	DATA	93 LITH	100
950	DATA	94 LITH	100
960	DATA	95 LITH	100
970	DATA	96 LITH	100
980	DATA	97 LITH	100
990	DATA	98 LITH	100
1000	DATA	99 LITH	100

END LABEL		CALL MPPR	
		LD R,R	
		BT	
AND R		.loop	
RET 1	parameters?	let address=i	
DEC A		LD R,0000	
IF 2,ptring	ptring only	PUSH BC	ptr counter
DEC A		BIT R,R	
IF 1,nopaper	get size	JP 1,label	
LD A,(I)+0		LD R,I	
LD label,A		.delay	
INC I		CALL frame	
INC I		LONG delay	
.nopaper		CALL roll	scroll
DEC R		.label	
IF 1,open	get paper	LD R,R	
LD A,(I)+0		LD HL,workspace	HL=workspace
LD (paper),A		.loop	BC=print addr
INC I		PUSH BC	
INC I		PUSH HL	save loop counter
.open		let paper=i	
LD A,(I)+0		LD C,BIT	
LD (pen),A	get pen	BLA HL,I	
INC I		IF C,set	get bit
INC I		let paper=i	
.ptring		LD C,R	
LD L,(I)+0		.set	
LD R,(I)+0	HL=descriptor	let address=i	
LD A,(HL)		LD R,R	
AND R		LD R,R	
RET 1	length=0?	LD L,I	get print address
LD (length),R		.here	
INC HL		LD (HL),I	
LD (address),HL		INC HL	
let address=i		LD (HL),R	
LD L,00000		DEC HL	
		LD A,R	
.scroll		ADD A,HL,LD R,A	
LD A,I		JP BC,ok	
CALL MPPR	page 1	LD R,ADDR	
LD HL,ADDR		ADD HL,DE	
LD (addr),HL		.ok	
LD HL,R		LONG here	
LD (offset),HL		LD R,R	
.letter		LD C,I	print line
LD A,(I)+0	get letter	POP HL	
CALL MPPR	get chr data address	INC HL	print data byte
CALL MPPR		POP BC	
LD R,workspace		LONG loop	
LD R,R		LD HL,(addr)	
LD R	transfer to workspace		

```

LD R0,44F
MNO A
DNC HL,BC
LD R0,0
Jloop0
LD HL,0,A
LD A,0
ADD A,00
LD R0,0
DNC loop0
LD HL,1edget
INC HL
LD 1edget,HL
POP BC
DNC loop1
INC R0
LD HL,length
DNC 0HL
JP R0,letter
RET
Jframe
PUSH BP
PUSH BC
LD R0,0F0
Jfr1

```

```

IN A,IC1
MNA
JH C,fr1
Jfr2
IN A,IC1
MNA
JH BC,fr2
POP BC
POP BP
RET

Jroll
LD HL,(offset)
INC HL
INC HL
LD 1offset,HL
MNA,0
MNA L
LD BC,0000
LD A,10
OUT (C),A
INC R0
LD A,0
ADD A,000
OUT (C),A

```

```

DNC R
LD A,10
OUT (C),A
INC R
LD A,L
OUT (C),A
RET

Joverlapan DEFS 0
Joffset DEFS 0
Jlength DEFS 0

END

```

offset/2