

```
FFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFF  
FFFFFFFFFFFFFFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF  
FFF
```

```
111  
111  
111  
111111  
111111  
111111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111111111  
111111111  
111111111
```

```
111  
111  
111  
111111  
111111  
111111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111  
111111111  
111111111  
111111111
```

```
AAAAAAAAAA  
AAAAAAAAAA  
AAAAAAAAAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAAAAAAAAAAAAAAA  
AAAAAAAAAAAAAAAA  
AAAAAAAAAAAAAAAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA  
AAA AAA
```



```

1 0001 0 MODULE RETDIR (
2 0002 0
3 0003 0     LANGUAGE (BLISS32),
4 0004 0     IDENT = 'V04-000'
5 0005 1 BEGIN
6 0006 1
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 *  ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 *  TRANSFERRED.
20 0020 1 *
21 0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 *  CORPORATION.
24 0024 1 *
25 0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1 **
32 0032 1
33 0033 1 FACILITY: F11ACP Structure Level 1
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1
37 0037 1     This routine returns the resulting data from a directory
38 0038 1     operation to the user's buffer packet.
39 0039 1
40 0040 1 ENVIRONMENT:
41 0041 1
42 0042 1     STARLET operating system, including privileged system services
43 0043 1     and internal exec routines. This routine must be called
44 0044 1     in kernel mode.
45 0045 1
46 0046 1 --
47 0047 1
48 0048 1
49 0049 1 AUTHOR: Andrew C. Goldstein, CREATION DATE: 3-Jan-1977 15:07
50 0050 1
51 0051 1 MODIFIED BY:
52 0052 1
53 0053 1     A0100 ACG0001 Andrew C. Goldstein, 10-Oct-1978 20:01
54 0054 1     Previous revision history moved to f11A.REV
55 0055 1
56 0056 1 **
57 0057 1

```



```

62 0375 1 GLOBAL ROUTINE RETURN_DIR (COUNT, STRING, ABD) : NOVALUE =
63 0376 1
64 0377 1 |++
65 0378 1
66 0379 1 FUNCTIONAL DESCRIPTION:
67 0380 1
68 0381 1     This routine returns the resulting data from a directory
69 0382 1     operation to the user's buffer packet.
70 0383 1
71 0384 1 CALLING SEQUENCE:
72 0385 1     RETURN_DIR (ARG1, ARG2, ARG3)
73 0386 1
74 0387 1 INPUT PARAMETERS:
75 0388 1     ARG1: byte count of result string
76 0389 1     ARG2: address of result string
77 0390 1     ARG3: address of buffer descriptors
78 0391 1
79 0392 1 IMPLICIT INPUTS:
80 0393 1     NONE
81 0394 1
82 0395 1 OUTPUT PARAMETERS:
83 0396 1     NONE
84 0397 1
85 0398 1 IMPLICIT OUTPUTS:
86 0399 1     NONE
87 0400 1
88 0401 1 ROUTINE VALUE:
89 0402 1     NONE
90 0403 1
91 0404 1 SIDE EFFECTS:
92 0405 1     result data written into buffer packet
93 0406 1
94 0407 1 |--
95 0408 1
96 0409 2 BEGIN
97 0410 2
98 0411 2 MAP
99 0412 2     STRING      : REF VECTOR [,BYTE], ! file string arg
100 0413 2     ABD         : REF BBLOCKVECTOR [,ABD$C_LENGTH];
101 0414 2                ! descriptor arg
102 0415 2
103 0416 2
104 0417 2 ! If the user provided a result length buffer, give him the length
105 0418 2 ! of the result string.
106 0419 2
107 0420 2
108 0421 2 IF .ABD[ABD$C_RESL, ABD$W_COUNT] GEQ 2
109 0422 2 THEN
110 0423 2     BEGIN
111 0424 2     (.ABD[ABD$C_RESL, ABD$W_TEXT] + ABD[ABD$C_RESL, ABD$W_TEXT] + 1) < 0, 16 > = .COUNT;
112 0425 2     END;
113 0426 2
114 0427 2 ! If the user provided a result string buffer, return as much of the
115 0428 2 ! result string as will fit (zero filling the buffer).
116 0429 2
117 0430 2
118 0431 2 CH$COPY (.COUNT, .STRING, 0,

```

```

: 119      0432 2      .ABD[ABD$C_RES, ABD$W_COUNT],
: 120      0433 2      .ABD[ABD$C_RES, ABD$W_TEXT] + ABD[ABD$C_RES, ABD$W_TEXT] + 1);
: 121      0434 2
: 122      0435 1 END;

```

. end of routine RETURN_DIR

```

.TITLE RETDIR
.IDENT \V04-000\

```

.PSECT \$CODE\$,NOWRT,2

```

: 0375
: 0421
: 0424
: 0433
: 0435

```

				003C 00000	.ENTRY RETURN DIR, Save R2,R3,R4,R5	
	52	0C	AC	D0 00002	MOVL ABD, R2	
	02	1A	A2	B1 00006	CMPW 26(R2), #?	
			0F	1F 0000A	BLSSU 1\$	
	51	18	A2	9E 0000C	MOVAB 24(R2), R1	
	50		01	3C 00010	MOVZWL (R1), R0	
		01	A140	9F 00013	PUSHAB 1(R1)[R0]	
	9E	04	AC	B0 00017	MOVW COUNT, @(SP)+	
	51	20	A2	9E 0001B 1\$:	MOVAB 32(R2), R1	
	50		61	3C 0001F	MOVZWL (R1), R0	
22	A2		00	08 BC	MOVCS COUNT, @STRING, #0, 34(R2), 1(R1)[R0]	
			04	AC 2C 00022		
			01	A140 0002A		
				04 0002D	RET	

: Routine Size: 46 bytes, Routine Base: \$CODE\$ + 0000

```

: 123      0436 1
: 124      0437 1 END
: 125      0438 0 ELUDOM

```

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	46	NOVEC,NOWRT, RD, EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	Symbols		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	6 0	1000	00:01.9

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:RETDIR/OBJ=OBJ\$:RETDIR MSRC\$:RETDIR/UPDATE=(ENH\$:RETDIR)

: Size: 46 code + 0 data bytes
: Run Time: 00:05.3
: Elapsed Time: 00:15.7
: Lines/CPU Min: 4939
: Lexemes/CPU-Min: 13601
: Memory Used: 63 pages
: Compilation Complete

The image displays a grid of 100 small terminal window screenshots, arranged in 10 rows and 10 columns. Each window shows a program name followed by 'LIS'. The programs are: Row 1: REQUEL LIS, RWATTR LIS; Row 2: MOOTFY LIS; Row 3: SCHFCB LIS; Row 4: MAKACC LIS; Row 5: MPWIND LIS; Row 6: MAPUBN LIS, PMS LIS, RDHEDR LIS, RWJB LIS; Row 7: RETDIR LIS; Row 8: ROBLOK LIS; Row 9: SMALOC LIS; Row 10: MAKMBE LIS, MAKSTR LIS, MXTHOR LIS. Each window contains various data, including text, numbers, and small graphical elements like bar charts or tables.