

RRRRRRRRRRRR		UUU		UUU	NNN		NNN	000000000		FFFFFFFFFFFFFF	FFFFFFFFFFFFFF
RRRRRRRRRRRR		UUU		UUU	NNN		NNN	000000000		FFFFFFFFFFFFFF	FFFFFFFFFFFFFF
RRRRRRRRRRRR		UUU		UUU	NNN		NNN	000000000		FFFFFFFFFFFFFF	FFFFFFFFFFFFFF
RRR	RRR	UUU		UUU	NNN		NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN		NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN		NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNNNNN		NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNNNNN		NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNNNNN		NNN	000	000	FFF	FFF
RRRRRRRRRRRR		UUU		UUU	NNN	NNN	NNN	000	000	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF
RRRRRRRRRRRR		UUU		UUU	NNN	NNN	NNN	000	000	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF
RRRRRRRRRRRR		UUU		UUU	NNN	NNN	NNN	000	000	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF
RRR	RRR	UUU		UUU	NNN	NNN	NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN	NNNNNN	NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN	NNNNNN	NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN	NNNNNN	NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN	NNN	NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN	NNN	NNN	000	000	FFF	FFF
RRR	RRR	UUU		UUU	NNN	NNN	NNN	000	000	FFF	FFF
RRR	RRR	UUUUUUUUUUUUUUUU		UUUUUUUUUUUUUUUU	NNN	NNN	NNN	000000000	000	FFF	FFF
RRR	RRR	UUUUUUUUUUUUUUUU		UUUUUUUUUUUUUUUU	NNN	NNN	NNN	000000000	000	FFF	FFF
RRR	RRR	UUUUUUUUUUUUUUUU		UUUUUUUUUUUUUUUU	NNN	NNN	NNN	000000000	000	FFF	FFF

```

LL          IIIIII      SSSSSSSS
LL          IIIIII      SSSSSSSS
LL          II         SS
LL          II         SS
LL          II         SS
LL          II         SS
LL          II         SSSSSS
LL          II         SSSSSS
LL          II         SS
LL          II         SS
LL          II         SS
LL          II         SS
LLLLLLLLLLL IIIIIIII   SSSSSSSS
LLLLLLLLLLL IIIIIIII   SSSSSSSS

```



```
1 0001 0 MODULE glnm ( IDENT = 'V04-000'
2 P 0002 0 %BLISS32[, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE,
3 0003 0 NONEXTERNAL = LONG_RELATIVE)]
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 1 * ALL RIGHTS RESERVED.
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 1 * TRANSFERRED.
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 1 * CORPORATION.
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
26 0026 1 *
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS
32 0032 1
33 0033 1 ABSTRACT: Get a list of numbers
34 0034 1
35 0035 1 ENVIRONMENT: Transportable
36 0036 1
37 0037 1 AUTHOR: R.W.Friday CREATION DATE: May, 1978
38 0038 1
39 0039 1
```

GLNM
V04-000

Revision History

1 7
16-Sep-1984 00:42:10
14-Sep-1984 13:06:36

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[RUNOFF.SRC]GLNM.BLI;1 Page (2)

:	41	0040	1	%SBTTL 'Revision History'
:	42	0041	1	
:	43	0042	1	MODIFIED BY:
:	44	0043	1	
:	45	0044	1	004 RER00004 Ron Randall 07-Mar-1983
:	46	0045	1	Global edit of all modules. Updated module names, idents,
:	47	0046	1	copyright dates. Changed require files to BLISS library.
:	48	0047	1	
:	49	0048	1	--
:	50	0049	1	

GNA
V04

: R

...


```
: 52      0050 1 %SBTTL 'Module Level Declarations'
: 53      0051 1
: 54      0052 1
: 55      0053 1  TABLE OF CONTENTS:
: 56      0054 1
: 57      0055 1  INCLUDE FILES:
: 58      0056 1
: 59      0057 1  LIBRARY 'NXPORT:XPORT';      ! XPORT Library
: 60      0058 1  REQUIRE 'REQ:RNODEF';        ! RUNOFF variant definitions
: 61      0189 1
: 62      U 0190 1  %IF DSRPLUS %THEN
: 63      U 0191 1  LIBRARY 'REQ:DPLLIB';        ! DSRPLUS BLISS Library
: 64      0192 1  %ELSE
: 65      0193 1  LIBRARY 'REQ:DSRLIB';        ! DSR BLISS Library
: 66      0194 1  %FI
: 67      0195 1
: 68      0196 1
: 69      0197 1  EXTERNAL REFERENCES:
: 70      0198 1
: 71      0199 1  EXTERNAL
: 72      0200 1      IRA : FIXED_STRING,
: 73      0201 1      KHAR,
: 74      0202 1      NMLST : NUMBER_LIST;
: 75      0203 1
: 76      0204 1  EXTERNAL ROUTINE
: 77      0205 1      GETNUM,
: 78      0206 1      RSKIPS;
: 79      0207 1
```

```

81 0208 1 GLOBAL ROUTINE GLNM (COUNT) : NOVALUE =
82 0209 1
83 0210 1 ++
84 0211 1 FUNCTIONAL DESCRIPTION:
85 0212 1
86 0213 1     Calls GETNUM to pick up "count" numbers, at most.
87 0214 1     Stops when a character is encountered that is not a comma,
88 0215 1     and GETNUM has previously returned a null value.
89 0216 1
90 0217 1 FORMAL PARAMETERS:
91 0218 1
92 0219 1     'COUNT' specifies the maximum number of numbers wanted.
93 0220 1
94 0221 1 IMPLICIT INPUTS:      None
95 0222 1
96 0223 1 IMPLICIT OUTPUTS:    None
97 0224 1
98 0225 1 ROUTINE VALUE:
99 0226 1 COMPLETION CODES:      None
100 0227 1
101 0228 1 SIDE EFFECTS:          None
102 0229 1 --
103 0230 1
104 0231 2 BEGIN
105 0232 2 LOCAL
106 0233 2     GETNUM_RESULT,
107 0234 2     KOUNT,
108 0235 2     NUMBER_SIGN,
109 0236 2     NUMBER_VALUE,
110 0237 2     NUMBER_LENGTH;
111 0238 2
112 0239 2 NMLST COUNT = 0;
113 0240 2 !If COUNT is greater than zero, then the list can be that long,
114 0241 2 !at most. If COUNT is less than or equal to zero, pick up as many
115 0242 2 !numbers as space is reserved for.
116 0243 2
117 0244 2 IF .COUNT GTR 0
118 0245 2 THEN
119 0246 2     KOUNT = .COUNT
120 0247 2 ELSE
121 0248 2     KOUNT = NMLST_MAXSIZE;
122 0249 2
123 0250 2 INCR I FROM 1 TO .KOUNT DO
124 0251 3 BEGIN
125 0252 3     GETNUM_RESULT = GETNUM (IRA, NUMBER_VALUE, NUMBER_SIGN, NUMBER_LENGTH);
126 0253 3     RSKIPS (IRA); !Skip spaces and tabs trailing a number.
127 0254 3
128 0255 3 IF .GETNUM_RESULT
129 0256 3 THEN
130 0257 4 BEGIN !Number was ok.
131 0258 4
132 0259 4 IF .NUMBER_LENGTH EQL 0
133 0260 4 THEN
134 0261 5 BEGIN
135 0262 5
136 0263 5 IF .KHAR EQL %C', '
137 0264 5 THEN
```



```
: 138      0265 6      BEGIN                                !It really is a null number, not the end of the list
: 139      0266 6      NMLST_COUNT = .I;
: 140      0267 6      NMLST_VALUE (.I) = 0;
: 141      0268 6      NMLST_DESCR (.I) = NM_NULL;
: 142      0269 6      KCNS ();                                !Skip the ','
: 143      0270 6      END
: 144      0271 6      !No more numbers in the list.
: 145      0272 5      ELSE
: 146      0273 5      RETURN;
: 147      0274 5
: 148      0275 5      END
: 149      0276 4      ELSE
: 150      0277 5      BEGIN                                !Not a null number
: 151      0278 5      RSKIPS (IRA);                        !Skip trailing spaces and tabs.
: 152      0279 5
: 153      0280 5      IF (.KHAR EQL %C',') AND
: 154      0281 6      (.I NEQ .KOUNT)                        !Don't skip any characters after the last item in the list
: 155      0282 5      THEN
: 156      0283 5      KCNS ();                                !Skip trailing comma.
: 157      0284 5
: 158      0285 5      NMLST_COUNT = .I;
: 159      0286 5      NMLST_VALUE (.I) = .NUMBER_VALUE;
: 160      0287 6      NMLST_DESCR (.I) = (IF .NUMBER_SIGN EQL 1
: 161      0288 6      THEN NM_PLUS ELSE IF .NUMBER_SIGN EQL -1
: 162      0289 5      THEN NM_MINUS ELSE NM_UNSIGNED);
: 163      0290 5      END
: 164      0291 5
: 165      0292 4      END
: 166      0293 3      ELSE
: 167      0294 4      BEGIN                                !Number in error.
: 168      0295 4      RSKIPS (IRA);                        !Skip trailing spaces and tabs.
: 169      0296 4
: 170      0297 4      IF (.KHAR EQL %C',') AND
: 171      0298 5      (.I NEQ .KOUNT)                        !Don't skip any characters after the last item in the list
: 172      0299 4      THEN
: 173      0300 4      KCNS ();                                !Skip trailing comma.
: 174      0301 4
: 175      0302 4      NMLST_COUNT = .I;
: 176      0303 4      NMLST_VALUE (.I) = .NUMBER_VALUE;
: 177      0304 4      NMLST_DESCR (.I) = NM_BAD;
: 178      0305 3      END;
: 179      0306 3
: 180      0307 2      END;
: 181      0308 2
: 182      0309 1      END;                                !End of GLNM
```

```
.TITLE  GLNM
.IDENT  \V04-000\

.EXTRN  IRA, KHAR, NMLST
.EXTRN  GETNUM, RSKIPS, RINTES

.PSECT  $CODE$,NOWRT,2
```

```
58 00000000G 01FC 00000
EF 9E 00002
```

```
.ENTRY  GLNM, Save R2,R3,R4,R5,R6,R7,R8
MOVAB   RSKIPS, R8
```

```
: 0208
:
```

57	00000000G	EF	9E	00009	MOVAB	KHAR, R7		
56	00000000G	EF	9E	00010	MOVAB	NMLST, R6		
55	00000000G	EF	9E	00017	MOVAB	IRA+12, R5		
5E		0C	C2	0001E	SUBL2	#12, SP		
		66	D4	00021	CLRL	NMLST		0239
	04	AC	D5	00023	TSTL	COUNT		0244
		06	15	00026	BLEQ	1\$		
53		04	AC	D0	00028	MOVL	COUNT, KOUNT	0246
		03	11	0002C	BRB	2\$		
53		28	D0	0002E	1\$: MOVL	#40, KOUNT		0248
		52	D4	00031	2\$: CLRL	I		0250
		4C	11	00033	BRB	7\$		
		5E	DD	00035	3\$: PUSHL	SP		0252
	08	AE	9F	00037	PUSHAB	NUMBER_SIGN		
	10	AE	9F	0003A	PUSHAB	NUMBER_VALUE		
	F4	A5	9F	0003D	PUSHAB	IRA		
00000000G	EF	04	FB	00040	CALLS	#4, GETNUM		
54		50	D0	00047	MOVL	RO, GETNUM_RESULT		
	F4	A5	9F	0004A	PUSHAB	IRA		0253
68		01	FB	0004D	CALLS	#1, RSKIPS		
03		54	E8	00050	BLBS	GETNUM_RESULT, 4\$		0255
	0080	31	00053	BRW	15\$			
	6E	D5	00056	4\$: TSTL	NUMBER_LENGTH			0259
	29	12	00058	BNEQ	8\$			
2C		67	D1	0005A	CMPL	KHAR, #44		0263
		01	13	0005D	BEQL	5\$		
		04	0005F	RET				
66		52	D0	00060	5\$: MOVL	I, NMLST		0266
	6642	D4	00063	CLRL	NMLST[I]			0267
	00A0	C642	D4	00066	CLRL	NMLST+160[I]		0268
		65	D5	0006B	TSTL	IRA+12		0269
		09	14	0006D	BGTR	6\$		
67	00G	8F	9A	0006F	MOVZBL	#RINTES, KHAR		
65		01	CE	00073	MNEGL	#1, IRA+12		
		5C	11	00076	BRB	14\$		
67	F8	B5	9A	00078	6\$: MOVZBL	@IRA+4, KHAR		
	F8	A5	D6	0007C	INCL	IRA+4		
		65	D7	0007F	DECL	IRA+12		
		51	11	00081	7\$: BRB	14\$		0263
	F4	A5	9F	00083	8\$: PUSHAB	IRA		0278
68		01	FB	00086	CALLS	#1, RSKIPS		
2C		67	D1	00089	CMPL	KHAR, #44		0280
		1B	12	0008C	BNEQ	10\$		
53		52	D1	0008E	CMPL	I, KOUNT		0281
		16	13	00091	BEQL	10\$		
		65	D5	00093	TSTL	IRA+12		0283
		09	14	00095	BGTR	9\$		
67	00G	8F	9A	00097	MOVZBL	#RINTES, KHAR		
65		01	CE	0009B	MNEGL	#1, IRA+12		
		09	11	0009E	BRB	10\$		
67	F8	B5	9A	000A0	9\$: MOVZBL	@IRA+4, KHAR		
	F8	A5	D6	000A4	INCL	IRA+4		
		65	D7	000A7	DECL	IRA+12		
66		52	D0	000A9	10\$: MOVL	I, NMLST		0285
6642	08	AE	D0	000AC	MOVL	NUMBER_VALUE, NMLST[I]		0286
01	04	AE	D1	000B1	CMPL	NUMBER_SIGN, #1		0287
		05	12	000B5	BNEQ	11\$		

50	02	D0	000B7	MOVL	#2, R0	
FFFFFFF 8F	12	11	000BA	BRB	13\$	0288
	04	AE	D1 000BC	11\$:	CMPL	NUMBER_SIGN, #-1
50	05	12	000C4	BNEQ	12\$	
	03	D0	000C6	MOVL	#3, R0	
50	03	11	000C9	BRB	13\$	
00A0 C642	01	D0	000CB	12\$:	MOVL	#1, R0
	50	D0	000CE	13\$:	MOVL	R0, NMLST+160[I]
	34	11	000D4	14\$:	BRB	18\$
	F4	A5	9F 000D6	15\$:	PUSHAB	IRA
68	01	FB	000D9	CALLS	#1, RSKIPS	0295
2C	67	D1	000DC	CMPL	KHAR, #44	0297
	1B	12	000DF	BNEQ	17\$	
53	52	D1	000E1	CMPL	I, KOUNT	0298
	16	13	000E4	BEQL	17\$	
	65	D5	000E6	TSTL	IRA+12	0300
	09	14	000E8	BGTR	16\$	
67	00G	8F	9A 000EA	MOVZBL	#RINTES, KHAR	
65	01	CE	000EE	MNEGL	#1, IRA+12	
	09	11	000F1	BRB	17\$	
67	F8	B5	9A 000F3	16\$:	MOVZBL	@IRA+4, KHAR
	F8	A5	D6 000F7	INCL	IRA+4	
	65	D7	000FA	DECL	IRA+12	
66	52	D0	000FC	17\$:	MOVL	I, NMLST
00A0 C642	08	AE	D0 000FF	MOVL	NUMBER_VALUE, NMLST[I]	0302
	04	D0	00104	MOVL	#4, NMLST+160[I]	0303
FF25 52	53	F1	0010A	18\$:	ACBL	KOUNT, #1, I, 3\$
	04	00110		RET		0304
						0250
						0309

; Routine Size: 273 bytes, Routine Base: \$CODE\$ + 0000

: 183 0310 1
: 184 0311 1 END
: 185 0312 0 ELUDOM

!End of module

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	273	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPI,ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
\$255\$DUA28:[SYSLIB]XPORT.L32;1	590	0	0	252	00:00.1
\$255\$DUA28:[RUNOFF.SRC]DSRLIB.L32;1	1248	20	1	86	00:00.3

GLNM
V04-000

Module Level Declarations

B 8
16-Sep-1984 00:42:10
14-Sep-1984 13:06:36

VAX-11 Bliss-32 V4.0-742
DISK\$VM\$MASTER:[RUNOFF.SRC]GLNM.BLI;1
Page 8
(4)

COMMAND QUALIFIERS

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

: Size: 273 code + 0 data bytes
: Run Time: 00:06.5
: Elapsed Time: 00:17.8
: Lines/CPU Min: 2893
: Lexemes/CPU-Min: 17731
: Memory Used: 79 pages
: Compilation Complete

0342

AH-BT13A-SE
 VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY