1. *Magic 8 Ball:* You are debugging an application in execution using gdb on a 64-bit (i.e. pointers use 64 bits), little-endian architecture. The application has a variable called *magic8ball* - defined as

char magic8ball[8][8][8];

Using gdb you find the following information at a particular stage in the application:

(gdb) p &magic8ball \$1 = (char (*)[8][8]] 0x7ffffffe000

And:

/	-75555	55-000						
(gdb) x/512bx 0x			0 76	0 65	0 70	0 00	0 00	0 00
0x7fffffffe0 <mark>00</mark> :		0x65	0x76	0x65	0x72	0×00	0×00	0×00
0x7fffffffe008:	0x4c	0x69	0x6b	0x65	0x6c	0x79	0x00	0×00
0x7fffffffe010:	0x4e	0x61	0x68	0x00	0x00	0×00	0x00	0×00
0x7fffffffe018:		0x6f	0x00	0xff	0xff	0x7f	0x00	0x00
0x7fffffffe020:		0x65	0x73	0x00	0x00	0x00	0x00	0x00
0x7fffffffe028:		0x61	0x68	0x00	0x00	0x00	0x00	0x00
0x7fffffffe030:	0x4d	0x61	0x79	0x62	0x65	0x00	0x00	0x00
0x7fffffffe038:	0x57	0x72	0x6f	0хбе	0x67	0x00	0x00	0x00
0x7ffffffffe040:	0x4d	0x61	0x79	0x62	0x65	0x00	0×00	0x00
0x7fffffffe048:	0x4e	0x65	0x76	0x65	0x72	0x00	0×00	0x00
0x7fffffffe050:	0x52	0x69	0x67	0x68	0x74	0x00	0×00	0x00
0x7fffffffe058:	0x57	0x72	0x6f	0хбе	0x67	0x00	0×00	0x00
0x7fffffffe060:	0x4e	0x65	0x76	0x65	0x72	0x00	$0 \times 0 0$	0x00
0x7fffffffe068:	0x4d	0x61	0x79	0x62	0x65	0x00	0×00	0x00
0x7fffffffe070:	0x4e	0x6f	0×00	0xff	0xff	0x7f	$0 \times 0 0$	0×00
0x7fffffffe078:	0x59	0x65	0x61	0x68	$0 \times 0 0$	0×00	$0 \times 0 0$	0x00
0x7ffffffffe0 <mark>80</mark> :	0x4c	0x69	0x6b	0×65	0х6с	0×79	0×00	0x00
0x7fffffffe088:	0x4e	0x61	0x68	0×00	$0 \times 0 0$	0×00	0×00	0x00
0x7fffffffe090:	0x4e	0x61	0x68	0×00	0×00	0×00	0×00	0x00
0x7fffffffe098:	0x59	0x65	0x73	0×00	0×00	0×00	0×00	0x00
0x7fffffffe0a0:	0x4c	0x69	0x6b	0x65	0x6c	0x79	0×00	0x00
0x7fffffffe0a8:	0x59	0x65	0x61	0x68	0x00	0×00	0×00	0x00
0x7fffffffe0b0:	0x4e	0x61	0x68	0×00	0×00	0×00	0×00	0x00
0x7fffffffe0b8:	0x59	0x65	0x61	0x68	0×00	0×00	0×00	0x00
0x7ffffffffe0 <mark>c0</mark> :	0x4d	0x61	0x79	0x62	0x65	0×00	0×00	0x00
0x7fffffffe0c8:	0x59	0x65	0x61	0x68	0x00	0x00	0×00	0x00
0x7fffffffe0d0:	0x4e	0x65	0x76	0x65	0x72	0×00	0×00	0x00
0x7fffffffe0d8:	0x57	0x72	0x6f	0хбе	0x67	0×00	0x00	0x00
0x7fffffffe0e0:	0x57	0x72	0x6f	0хбе	0x67	0x00	0×00	0x00
0x7fffffffe0e8:	0x4e	0x6f	0x00	0xff	0xff	0x7f	0×00	0x00
0x7fffffffe0f0:	0x4d	0x61	0x79	0x62	0x65	0x00	0×00	0x00
0x7fffffffe0f8:	0x57	0x72	0x6f	0хбе	0x67	0x00	0×00	0x00
0x7fffffffe100:	0x4e	0x61	0x68	0×00	0x00	0x00	0×00	0x00
0x7fffffffe108:	0x52	0x69	0x67	0x68	0x74	0x00	0×00	0x00
<pre>0x7ffffffffe110:</pre>	0x53	0x75	0x72	0×65	0×00	0×00	0×00	0x00
0x7ffffffffe118:	0x4e	0x6f	0×00	0xff	0xff	0x7f	0×00	0x00
0x7ffffffffe120:	0x59	0x65	0x61	0x68	0×00	0×00	0×00	0x00
0x7ffffffffe128:	0x4e	0x61	0x68	0x00	0x00	0x00	0x00	0x00
0x7fffffffe130:	0x4e	0x61	0x68	0x00	0×00	0x00	0x00	0x00
0x7ffffffffe138:	0x52	0x69	0×67	0x68	0×74	0x00	0×00	0x00
0x7ffffffffe140:	0x59	0x65	0x73	0x00	0×00	0x00	0×00	0x00
0x7ffffffffe148:	0x59	0x65	0x61	0x68	0×00	0x00	0x00	0x00
0x7ffffffffe150:	0x4e	0x6f	0×00	0xff	0xff	0x7f	0x00	0x00

0x7fffffffe168: 0x4c	002
0x7fffffffe170: 0x52	000
0x7ffffffffe178: 0x4e $0x6f$ $0x00$ $0xff$ $0xff$ $0x7f$ $0x00$ $0x$	00

0x7fffffffe180: 0x4e 0x65 0x76 0x65 0x72 0x00 0x00 0x	- 0 0
	100
0x7fffffffe188: 0x57	002
0x7fffffffe190: 0x59	002
0x7fffffffe198: 0x4e 0x65 0x76 0x65 0x72 0x00 0x00 0x	002
0x7fffffffela0: 0x4e	002
0x7fffffffe1a8: 0x4c 0x69 0x6b 0x65 0x6c 0x79 0x00 0x	002
0x7fffffffe1b0: 0x53	002
0x7fffffffe1b8: 0x59	002
0x7fffffffe1c0: 0x53	002
0x7fffffffe1c8: 0x53	002
0x7fffffffe1d0: 0x4e	002
0x7fffffffeld8: 0x4c	002
0x7fffffffe1e0: 0x4d 0x61 0x79 0x62 0x65 0x00 0x00 0x	002
0x7fffffffe1e8: 0x57	002
0x7fffffffe1f0: 0x53	002
0x7fffffffe1f8: 0x4c 0x69 0x6b 0x65 0x6c 0x79 0x00 0x	

If the application were to output the value of magic8ball[3][4] – what would it be? i.e. what would be returned from the statement printf("%s", magic8ball[3][4]);

ASCII Table

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	0	96	60	,
1	1	Start of heading	SOH	CTRL-A	33	21	1	65	41	Α	97	61	a
2	2	Start of text	STX	CTRL-B	34	22		66	42	В	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	С
4	4	End of xmit	EOT	CTRL-D	36	24	\$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	е
6	6	Acknowledge	ACK	CTRL-F	38	26	8.	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27		71	47	G	103	67	9
8	8	B ackspace	BS	CTRL-H	40	28	(72	48	Н	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29)	73	49	I	105	69	i
10	OA.	Line feed	LF	CTRL-J	42	2A		74	4Α.	J	106	64	j
11	OB	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	OC.	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	1
13	OD	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	М	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E		78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	0	111	6F	0
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	р
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	Т	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	٧
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	×
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	у
26	1A	Substitute	SUB	CTRL-Z	58	ЗА	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[59	3B	;	91	5B	[123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	1
29	1D	Group separator	GS	CTRL-]	61	3D	-	93	5D]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL	63	3F	?	95	5F	_	127	7F	DEL