Consider function "check", which is designed to return 1 if its argument is an ASCII character which is a decimal digit (and 0 otherwise).

The assembly listing for that function is shown below:

```
1 .global check
2 .text
3 .balign 4
4 check:
5 0000 E52DE004    push    {lr}  @ Make copy of incoming character
6
7 0004 E1A01000    mov     r1, r0    @ Make copy of incoming character
8
9 0008 E3A00000    mov     r0, #0    @ Assume return value is false
10 000c E3510030    cmp     r1, #'0'  @ Compare character to '0'
11 0010 BA XXXXX    blt     endif  @ Return value is true
12 0014 E3510039    cmp     r1, #'9'  @ Compare character to '9'
13 0018 CA XXXXX    bgt     endif
14 001c E3A00001    mov     r0, #1    @ Return value is true
15 endif:
16
17 0020 E49DE004    pop     {lr}
18 0024 E12FFF1E    bx      lr
```

a) Complete the symbol table which would be produced by the assembler.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>A/R</th>
<th>L/G</th>
<th>Definition</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>check</td>
<td>line 4</td>
<td></td>
<td></td>
<td>none</td>
<td></td>
</tr>
<tr>
<td>endif</td>
<td>line 15</td>
<td>.text+10, .text+18</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

b) Complete the machine language instruction at line 11 (replace XXXXX with the correct hexadecimal value). Show your work.

```
11 0010 BA XXXXX    blt     endif
```

```
11 0010 E3000000    blt     endif
```

c) Complete the machine language instruction at line 13 (replace XXXXX with the correct hexadecimal value). Show your work.

```
13 0018 CA XXXXX    bgt     endif
```

```
13 0018 E3000000    bgt     endif
```