



[< Previous](#)



[Next >](#)

# WE10.1

[🔖 Bookmark this page](#)

Video explanation of solution is provided below the problem.

For all Beta related questions, you should make use of the [Beta documentation](#), the [Beta Instruction Summary](#), and the [Beta Diagram](#).

### Beta Assembly

7/7 points (ungraded)

For each of the Beta instruction sequences shown below, indicate the values of the specified quantities after the sequence has been executed. Consider each sequence separately and assume that execution begins at location 0 and halts when the HALT() instruction is about to be executed. Also assume that all registers have been initialized to 0 before execution begins. Remember that even though the Beta loads and stores 32-bit words from memory, all addresses are *byte addresses*, i.e., the addresses of successive words in memory differ by 4.

Fill in requested values left after execution of each segment, or “CAN'T TELL” where appropriate.

1.

```
. = 0
LD(R31, c, R0)
ADDC(R0, b, R0)

HALT()

. = 0x200
a: LONG(0x100)
b: LONG(0x200)
c: LONG(0x300)
```

Value left in R0 (HEX): 0x

504

✓

Value assembler assigns to the symbol “c”: 0x

208

✓

2.

```
. = 0
BR(. + 4, R0)
HALT()
```

Value left in R0: 0x

4

✓

3.

```
. = 0
LD(R31, x, R0)
CMOVE(0, R1)

loop: ANDC(R0, 1, R3)
      ADD(R3, R1, R1)
      SHRC(R0, 1, R0)
      BNE(R0, loop)
      HALT()

x:    LONG(0x0FACE0FF)
```

Value left in R0: 0x

0

✓

Value left in R1: 0x

13

✓

4.

```
. = 0
CMOVE(0x1000, SP)
PUSH(SP)
```

Calculator

```
HALT ( )
```

Value left in SP (HEX): 0x  ✓

Value pushed onto stack (HEX): 0x  ✓

Submit

Beta Assembly



Beta Assembly

```
. = 0
BR(. + 4, R0)
HALT()
```

- Value left in R0 (HEX)?



(Caption will be displayed when you start playing the video.)



▶ 0:00 / 0:00

▶ 1.0x

🔊

🗑️

CC

🗣️

Video

[Download video file](#)

Transcripts

[Download SubRip \(.srt\) file](#)

[Download Text \(.txt\) file](#)

Discussion

Hide Discussion

Topic: 10. Assembly Language, Models of Computation / WE10.1

Add a Post

Show all posts ▼by recent activity ▼

💬 [The code does not count the number of 1's.](#)

[The code doesn't count the number of 1's. Please check. The pseudocode code is: 1.  \$r0 = 0x0FACE0FF/LD\(R31, x, R0\)\$  2.  \$r1\$](#)  5

Calculator

< Previous

Next >

© All Rights Reserved



# edX

- [About](#)
- [Affiliates](#)
- [edX for Business](#)
- [Open edX](#)
- [Careers](#)
- [News](#)

# Legal

- [Terms of Service & Honor Code](#)
- [Privacy Policy](#)
- [Accessibility Policy](#)
- [Trademark Policy](#)
- [Sitemap](#)

# Connect

- [Blog](#)
- [Contact Us](#)
- [Help Center](#)
- [Media Kit](#)
- [Donate](#)



© 2021 edX Inc. All rights reserved.  
深圳市恒宇博科技有限公司 [粤ICP备17044299号-2](#)