CSE 3666 - Lab 3
Jonathan Ameri

```
1   #        CSE 3666 Lab 3: remove spaces
2
3           .data
4           .align 2
5           # allocating space for both strings
6   str:    .space  128
7   res:    .space  128
8
9           .globl  main
10
11          .text
12  main:
13
14          # read a string into str
15          # use pseudoinstruction la to load address into register
16          la      a0, str
17          li      a1, 100 # a1 is the max number of characters to read
18                          # the maximum number of characters for input is 100
19          li      a7, 8
20          ecall
21
22          # a0 is the address of str
23          la      a1, res
24
25
```

```
25
26
27          addi t0, a0, 0          # t0 = i = a0    [t0 is temporary register for indexing]
28          addi t1, a1, 0          # t1 = j = a1    [t1 is temporary register for indexing]
29          addi a2, x0, 32         # a2 = 32
30  loop:   lb t2, 0(t0)           # load byte into t2 [t2 is used to see if 'current' byte is a space
31          beq t2, a2, skip       # if the current byte is a space, don't write to res
32          sb t2, 0(t1)           # if the current byte is NOT a space, write that byte to res
33          addi t1, t1, 1         # increment i
34  skip:   addi t0, t0, 1         # increment j
35          bne t2, x0, loop       # if we reach null terminator, end the loop
36
37          # load result into a0 and print result
38          li a7, 4
39          la a0, res
40          ecall
41
42
43          # TODO
44          # remove spaces in str
45          # print res
46
47  exit:   addi    a7, x0, 10
48          ecall
49
```

Describe the results of the code. Does the code work or only work for some inputs? Choose some strings yourself and show the program works correctly. Include a screenshot of "Run I/O" tab showing a few runs of the code. Include some strings you have selected.

The code works as intended. You can input a string with spaces and the program will print back the string without any spaces included. The limitations of the program are that you cannot type in a string longer than 99 characters because the address of a1 is set to be 100 bytes long. When a

99 character string is used, the result is not printed on a newline, because the newline character was never recorded.

```
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-- program is finished running (0) --
```

```
1 2 3 4 5
12345

-- program is finished running (0) --
```

```
hi my name is jon
himynameisjon

-- program is finished running (0) --
```

```
i love cse 3666
ilovecse3666

-- program is finished running (0) --
```