User Id : weizhe.goh@digipen.edu	Started : 2019.11.21 17:20:59	Submitted : 2019.11.26 11:27:34	Score : 100

DigiPen CS Vadim Surov CS100A Assignment #9

Assembler - Arrays

Practice to use assember to work with null terminated strings.

Rules

Read carefully and check all rules you agree with:

■ Each card has description which must be strictly followed.

■ Keep the code properly formatted (correct indentation, line width is 40 characters max, no empty lines).

■ Your code must represent your own individual work. If something is not clear, ask your instructor for help.

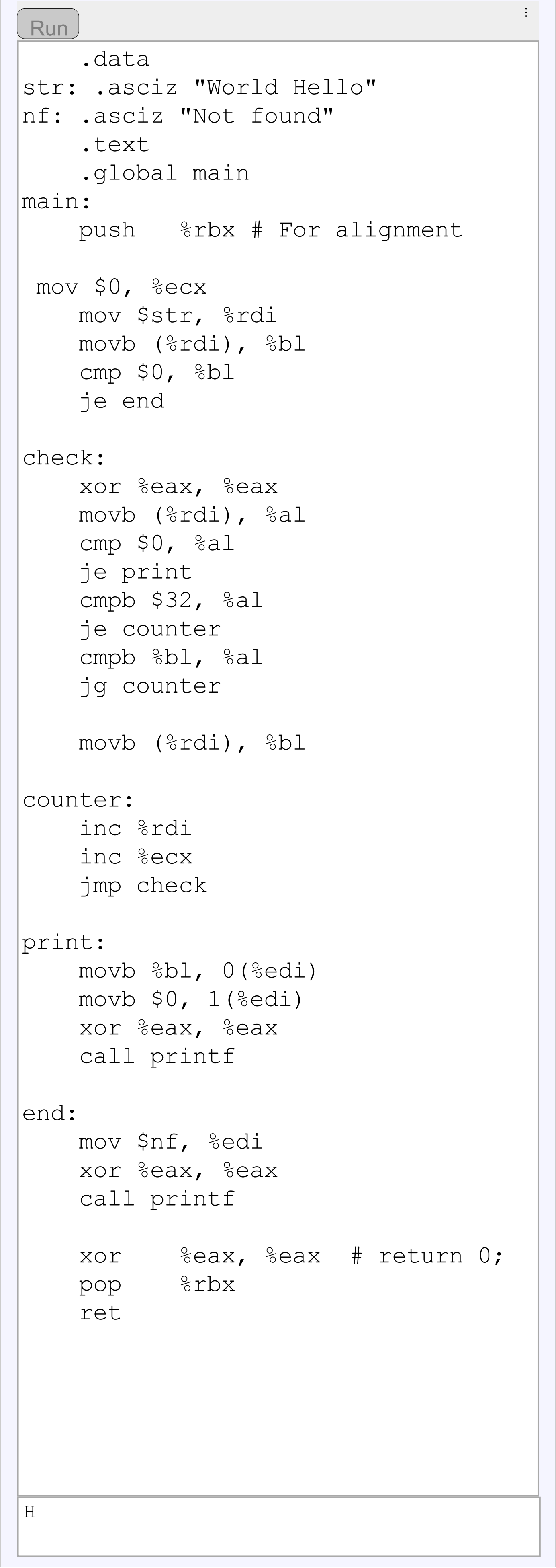
■ Cheating of any kind (copying someone else's work, allowing others to copy your work, collaborating, etc.) will not be tolerated and will be dealt with SEVERELY.

Problem

- Given a null-terminated string as str. The string can have any length.
- Create the code that determines the smallest character in the string excluding whitespaces (characters with code 32).
- You code must output the character or "Not found" when string is empty.
- Place your the same code in the middle of the following test cases. Do not modify given values in the tests.
- It's not allowed to use any standard library function except printf.

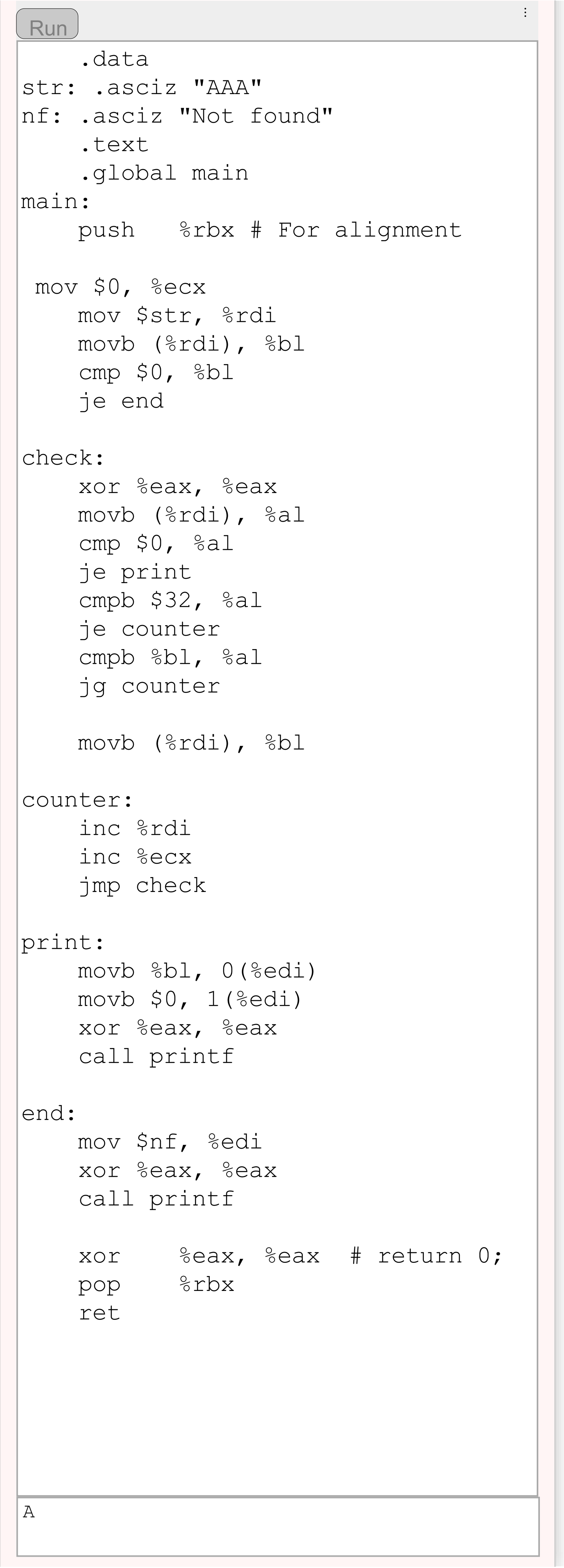
★ Test 1 1 ☑

★ Test 2 1 ☑



```
    .data
str: .asciz "World Hello"
nf: .asciz "Not found"
    .text
    .global main
main:
    push   %rbx # For alignment

 mov $0, %ecx
    mov $str, %rdi
    movb (%rdi), %bl
    cmp $0, %bl
    je end

check:
    xor %eax, %eax
    movb (%rdi), %al
    cmp $0, %al
    je print
    cmpb $32, %al
    je counter
    cmpb %bl, %al
    jg counter

    movb (%rdi), %bl

counter:
    inc %rdi
    inc %ecx
    jmp check

print:
    movb %bl, 0(%edi)
    movb $0, 1(%edi)
    xor %eax, %eax
    call printf

end:
    mov $nf, %edi
    xor %eax, %eax
    call printf

    xor    %eax, %eax  # return 0;
    pop    %rbx
    ret
```

```
H
```



```
    .data
str: .asciz "AAA"
nf: .asciz "Not found"
    .text
    .global main
main:
    push   %rbx # For alignment

 mov $0, %ecx
    mov $str, %rdi
    movb (%rdi), %bl
    cmp $0, %bl
    je end

check:
    xor %eax, %eax
    movb (%rdi), %al
    cmp $0, %al
    je print
    cmpb $32, %al
    je counter
    cmpb %bl, %al
    jg counter

    movb (%rdi), %bl

counter:
    inc %rdi
    inc %ecx
    jmp check

print:
    movb %bl, 0(%edi)
    movb $0, 1(%edi)
    xor %eax, %eax
    call printf

end:
    mov $nf, %edi
    xor %eax, %eax
    call printf

    xor    %eax, %eax  # return 0;
    pop    %rbx
    ret
```

```
A
```

★	Test 3	1	☑

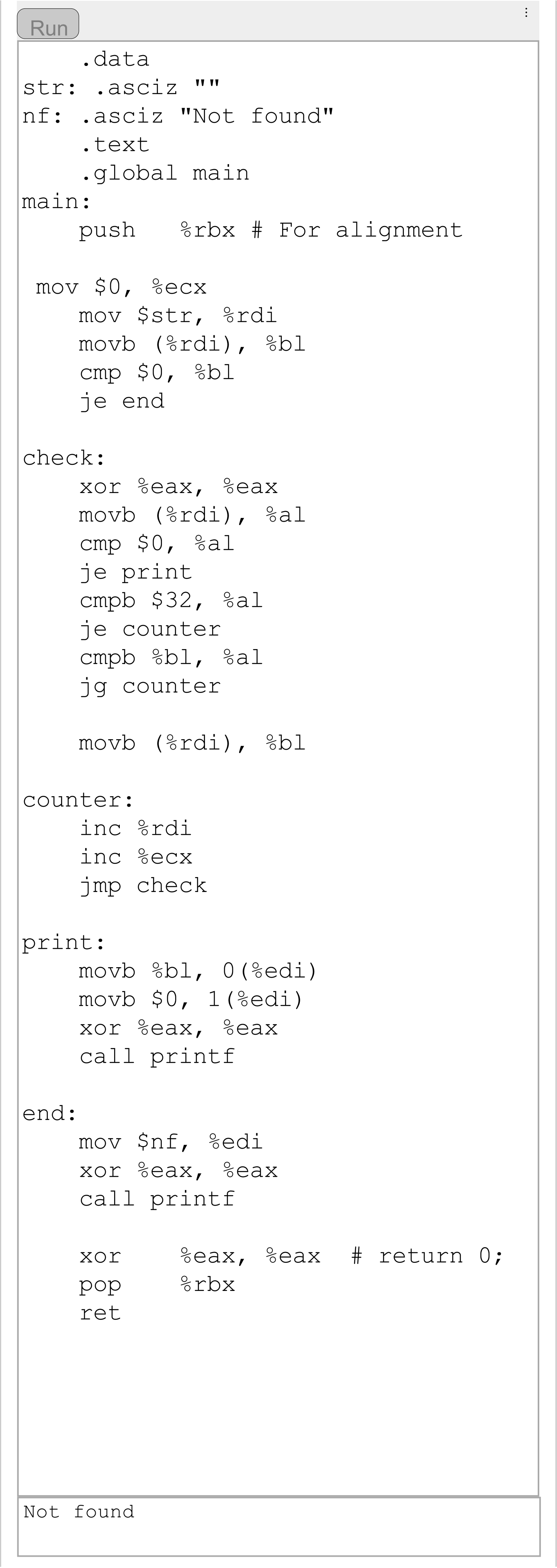
Survey

- What is approximate number of hours you spent implementing this assignment?

```
2day
```

- Indicate the specific portions of the assignment that gave you the most trouble

```
compare and return smallest
character
```



```
    .data
str: .asciz ""
nf: .asciz "Not found"
    .text
    .global main
main:
    push   %rbx # For alignment

 mov $0, %ecx
    mov $str, %rdi
    movb (%rdi), %bl
    cmp $0, %bl
    je end

check:
    xor %eax, %eax
    movb (%rdi), %al
    cmp $0, %al
    je print
    cmpb $32, %al
    je counter
    cmpb %bl, %al
    jg counter

    movb (%rdi), %bl

counter:
    inc %rdi
    inc %ecx
    jmp check

print:
    movb %bl, 0(%edi)
    movb $0, 1(%edi)
    xor %eax, %eax
    call printf

end:
    mov $nf, %edi
    xor %eax, %eax
    call printf

    xor    %eax, %eax  # return 0;
    pop    %rbx
    ret
```

```
Not found
```

By signing this document you fully agree that all information provided therein is complete and true in all respects.	Responder sign:

Copyright © 2019 | Powered by MyTA | www.mytaonline.com