

B.2 Εξοικείωση με την MIPS Assembly (A) Αλλάξτε τον παραπάνω κώδικα ώστε να ζητάει να εισαχθεί ένα string (your_string) και στη συνέχεια να εκτυπώνει Hello your_string World!

Προφανώς πλέον το Hello, και το World! είναι δύο διαφορετικά strings.

lab1b2 - Σημειωματάριο

Αρχείο Επεξεργασία Μορφή Προβολή Βοήθεια

B2

Data declaration section

#

.data

#Strings to be printed

#

out_string1: .ascii "\nHello, "

your_string: .space 20

out_string2: .ascii "World! \n"

ask_string: .ascii "\nEnter a 20-character string: \n"

Assembly language instructions go in text segment

#

.text

Start of code section

#

main:

#Asks for string from user

#

li \$v0, 4

la \$a0, ask_string

syscall

#Get user's input as text

#

li \$v0, 8

la \$a0, your_string

li \$a1, 20

syscall

#Prints Hello

#

li \$v0, 4

la \$a0, out_string1

syscall

#Prints user's input

#

li \$v0, 4

la \$a0, your_string

syscall

#Prints World

#

li \$v0, 4

la \$a0, out_string2


syscall

#Terminates program

#

li \$v0, 10

syscall

 Console

— □ ×

```
Enter a 20-character string:  
beautiful
```

```
Hello, beautiful  
World!
```

B.3 Εξοικείωση με την MIPS Assembly (B) Αλλάξτε τον παραπάνω κώδικα ώστε να ζητάει να εισαχθεί ένας αριθμός (integer) και στη συνέχεια να εκτυπώνει Hello World! (2* integer)

Προσοχή εδώ πρέπει να αποφασίσετε αν το Hello, και το World! είναι ακόμα δύο διαφορετικά strings.

Στην περίπτωση αυτή το Hello World ! μπορεί να είναι ένα string.



lab1b3 - Σημειωματάριο

Αρχείο Επεξεργασία Μορφή Προβολή Βοήθεια

```
# B3

# Data declaration section
#
.data

# Strings to be printed
#
f_string: .ascii "\nEnter a integer:\n"
s_string: .ascii "\nHello World!"
t_string: .ascii "( "
fo_string: .ascii " )\n"

# Assembly language instructions go in text segment
#
.text

# Start of code section
#
main:

# Asks for integer from user
#
li $v0, 4
la $a0, f_string
syscall

# Gets user's input as an integer
#
li $v0, 5
syscall

# Stores $t0<-$v0
#
add $t0, $v0, $0

# Prints Hello World
#
li $v0, 4
la $a0, s_string
syscall

#Prints "("
#
li $v0, 4
la $a0, t_string
syscall

#Doubles the integer
#
li $v0, 1
sll $a0, $t0, 1
syscall

#Prints ")"
#
li $v0, 4
la $a0, fo_string
syscall

#Terminates program
#
li $v0, 10
syscall
```

Console

Enter a integer:
10

Hello World! (20)

B.4 Εξοικείωση με την MIPS Assembly (Γ) Αλλάξτε τον παραπάνω κώδικα ώστε να ζητάει να εισαχθεί ένας χαρακτήρας (character) και στη συνέχεια να εκτυπώνει Hello World!(Character)

lab1b4 - Σημειωματάριο

Αρχείο Επεξεργασία Μορφή Προβολή Βοήθη

```
#B4

# Data declaration section
#
.data

#Strings to be printed
#
out_string: .asciiiz "\nHello World!"
ask_char: .asciiiz "\nEnter a character:\n"
t_string: .asciiiz "( "
f_string: .asciiiz " )\n"

# Assembly language instructions go in text segment
#
.text

# Start of code section
#

main:

#Asks for character from user
#
li $v0, 4
la $a0, ask_char
syscall

#Gets user's input as a character
#
li $v0, 12
syscall

#Stores the character in $t0
#
move $t0, $v0


#Prints Hello World!
#
li $v0, 4
la $a0, out_string
syscall

#Prints "("
#
li $v0, 4
la $a0, t_string
syscall

#Prints user's input
#
li $v0, 11
move $a0, $t0
syscall

#Prints ")"
#
li $v0, 4
la $a0, f_string
syscall

#Terminates program
#
li $v0, 10
syscall
```

 Console

— □ ×

Enter a character:

g

Hello World! (g)