**Bahria University, Lahore Campus**

Department of Computer Sciences

Lab Journal 014

**(Spring 2023)**

|  |  |  |
| --- | --- | --- |
| Course: | **Computer Architecture & Organization Lab** |  |
| Course Code: | CEL 221 | Max Marks: 30 |
| Faculty’s Name: | Maryam Munawar | Lab Engineer: |

Name: \_AFFAN AHMAD\_\_\_ Enroll No: \_\_03-134221-003\_\_

### Task1: 30 Minutes

### Write a program to compare two strings whether they are exactly equal or not.

.data

str1: .asciiz "Hello"

str2: .asciiz "hello"

equalMsg: .asciiz "The strings are equal."

notEqualMsg: .asciiz "The strings are not equal."

.text

.globl main

main:

la $a0, str1

la $a1, str2

jal compareStrings

beqz $v0, equal

li $v0, 4

la $a0, notEqualMsg

syscall

j exit

equal:

li $v0, 4

la $a0, equalMsg

syscall

exit:

li $v0, 10

syscall

compareStrings:

lb $t0, 0($a0)

lb $t1, 0($a1)

compareLoop:

beqz $t0, compareEnd

beq $t0, $t1, continue

li $v0, 4

la $a0, notEqualMsg

syscall

li $v0, 0

jr $ra

continue:

addiu $a0, $a0, 1

addiu $a1, $a1, 1

lb $t0, 0($a0)

lb $t1, 0($a1)

j compareLoop

compareEnd:

beqz $t0, equalResult

li $v0, 4

la $a0, notEqualMsg

syscall

li $v0, 0

jr $ra

equalResult:

li $v0, 1

jr $ra

### .data

### str1: .asciiz "Hello"

### str2: .asciiz "hello"

### equalMsg: .asciiz "The strings are equal."

### notEqualMsg: .asciiz "The strings are not equal."

### 

### .text

### .globl main

### main:

### la $a0, str1

### la $a1, str2

### jal compareStrings

### beqz $v0, equal

### li $v0, 4

### la $a0, notEqualMsg

### syscall

### j exit

### 

### equal:

### li $v0, 4

### la $a0, equalMsg

### syscall

### 

### exit:

### li $v0, 10

### syscall

### compareStrings:

### lb $t0, 0($a0)

### lb $t1, 0($a1)

### 

### compareLoop:

### beqz $t0, compareEnd

### beq $t0, $t1, continue

### 

### 

### li $v0, 4

### la $a0, notEqualMsg

### syscall

### li $v0, 0

### jr $ra

### 

### continue:

### addiu $a0, $a0, 1

### addiu $a1, $a1, 1

### lb $t0, 0($a0)

### lb $t1, 0($a1)

### j compareLoop

### 

### compareEnd:

### beqz $t0, equalResult

### 

### li $v0, 4

### la $a0, notEqualMsg

### syscall

### li $v0, 0

### jr $ra

### 

### equalResult:

### li $v0, 1

### jr $ra

### 

### Task2: 30 Minutes

### Write a program to check a string is palindrome or not.

### .data

### str: .asciiz "madam"

### palindrome: .asciiz "The string is a palindrome."

### not\_palindrome: .asciiz "The string is not a palindrome."

### .text

### .globl main

### main:

### la $a0, str

### jal palindrome\_check

### 

### 

### beqz $v0, not\_palindrome

### li $v0, 4

### la $a0, palindrome

### syscall

### j exit

### palindrome\_check:

### move $t0, $a0

### 

### # Determine the length of the string

### li $t1, 0

### count\_length:

### lb $t2, ($t0)

### beqz $t2, compare\_palindrome

### 

### addi $t1, $t1, 1

### addi $t0, $t0, 1

### j count\_length

### 

### compare\_palindrome:

### subi $t1, $t1, 1

### 

### move $t2, $a0

### add $t2, $t2, $t1

### 

### move $t3, $t0

### 

### check\_palindrome:

### lb $t4, ($t3)

### lb $t5, ($t2)

### bne $t4, $t5, not\_palindrome

### 

### addi $t3, $t3, 1

### subi $t2, $t2, 1

### 

### bge $t3, $t2, palindrome\_found

### j check\_palindrome

### palindrome\_found:

### li $v0, 1

### jr $ra

### exit:

### li $v0, 10

### syscall

### Task3: 30 Minutes

### Write a program to convert a string in uppercase.

**.data**

**str: .asciiz "Hello, World!"**

**.text**

**.globl main**

**main:**

**la $a0, str**

**jal convert\_to\_uppercase**

**li $v0, 4**

**move $a0, $t0**

**syscall**

**j exit**

**convert\_to\_uppercase:**

**move $t0, $a0**

**loop:**

**lb $t1, ($t0)**

**beqz $t1, end\_conversion**

**addi $t1, $t1, -32**

**sb $t1, ($t0)**

**addi $t0, $t0, 1**

**j loop**

**end\_conversion:**

**jr $ra**

**exit:**

**li $v0, 10**

**syscall**