**Bahria University, Lahore Campus**

Department of Computer Sciences

Lab Journal 012

**(Spring 2023)**

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

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enroll No: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Lab Tasks:

### Task1: 10 Minutes

The data structure for this new stringStack class is defined below.

class stringStack

{

int SIZE=7;

int elements[SIZE]; # Note that characters are stored as int

int last = SIZE-1;

push(String s) {

last = (last - s.length())-1;

elements[last] = s.length();

int i = last + 1;

for (char c in s)

{

elements[i] = c;

i = i + 1;

}

}

String pop()

{

int i = elements[last];

int j = last + 1;

last = last + i;

for ( ; j < last; j++) {

s = s + elements[j];

}

return s;

}

}

### Task2: 10 Minutes

subprogram global main()

{

register int multiplicand

register int multiplier

register int answer

m = prompt("Enter the multiplicand")

n = prompt("Enter the multiplier")

answer = Multiply(m, n)

print("The answer is: " + answer)

}

subprogram int multiply(int m, int n)

{

if (n == 1)

return m;

return m + multiply(m,n-1)

}

Solution:

.text

.globl main

main:

# register conventions

# $s0 - m

# $s1 - n

# $s2 - answer

la $a0, prompt1 # Get the multiplicand

jal PromptInt

move $s0, $v0

la $a0, prompt2 # Get the multiplier

jal PromptInt

move $s1, $v0

move $a0, $s0

move $a1, $s1

jal Multiply # Do multiplication

move $s2, $v0

la $a0, result #Print the answer

move $a1, $s2

jal PrintInt

jal Exit

Multiply:

addi $sp, $sp -8 # push the stack

sw $a0, 4($sp) #save $a0

sw $ra, 0($sp) # Save the $ra

seq $t0, $a1, $zero # if (n == 0) return

addi, $v0, $zero, 0 # set return value

bnez $t0, Return

addi $a1, $a1, -1 # set n = n-1

jal Multiply # recurse

lw $a0, 4($sp) # retrieve m

add $v0, $a0, $v0 # return m+multiply(m, n-1)

Return:

lw $ra, 0($sp) #pop the stack

addi $sp, $sp, 8

jr $ra

.data

prompt1: .asciiz "Enter the multiplicand: "

prompt2: .asciiz "Enter the multiplier: "

result: .ascii "The answer is: "

.include "utils.asm"

### Task3: 10 Minutes

Write a recursive program to calculate factorial numbers. Use the definition of factorial as F(n) = n \* F(n-1)

**Lab Grading Sheet :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Max Marks** | **Obtained Marks** | **Comments(*if any*)** |
| a. | 10 |  |  |
| b. | 10 |  |  |
| c. | 10 |  |  |
|  |  |  |  |
| **Total** | **30** |  | **Signature** |

**Note : Attempt all tasks and get them checked by your Instructor**