Bahria University,

Karachi Campus

LAB EXPERIMENT NO.

04

LIST OF TASKS

TASK NO	OBJECTIVE
1.	Answer the following questions related to the MARS MIPS stimulator after executing the following example: a. Where (to which window) is the output data displayed? b. Write down the address of the first instruction of the program (see the text window). c. Write down the value of the register $sp before and after executing the program. d. Write down the values of $a0 and $v0 before and after the execution of program in Register window and explain the logic behind the changing in values?
2.	Write an assembly program that prints the name of your favorite book, quote, couplet, movie, drama, color and dish.
3.	Write an assembly program that takes the student information like name, enrollment, registration number, program, semester and section from user and then print it in proper format for ID Card.

Submitted On:

Date: 03/10/2025

Task No. 01:

Answer the following questions related to the MARS MIPS stimulator after executing the following example:

a. Where (to which window) is the output data displayed?

b. Write down the address of the first instruction of the program (see the text window).

c. Write down the value of the register $sp before and after executing the program.

d. Write down the values of $a0 and $v0 before and after the execution of program in Register window and explain the logic behind the changing in values?

```
.data
input_prompt: .asciiz "Enter an integer: "

.text
main:
li $v0, 4
la $a0, input_prompt
syscall

li $v0, 5
syscall
move $t0, $v0

move $a0, $t0
li $v0, 1
syscall

li $v0, 10
syscall
```

Solution:

a. Where (to which window) is the output data displayed?

   Ans. The output is viewed at the console window of MARS.

b. Write down the address of the first instruction of the program (see the text window).

   Ans. The address of the first instruction is 0x00400000.

c. Write down the value of the register $sp before and after executing the program.

   Ans. The value of register $sp before execution: 0x7fffeffc and after execution: 0x7fffeffc as $sp is not used in the program

   Write down the values of $a0 and $v0 before and after the execution of program in Register window and explain the logic behind the changing in values?

d. Write down the values of $a0 and $v0 before and after the execution of program in Register window and explain the logic behind the changing in values?

   Ans. Initially the values are $a0 = 0, $v0 = 0
   After execution $a0 = entered number, $v0 first: 4(for displaying string), second: 5 (to take input), third: entered number, forth: 1 (display integer), last: a (hexadecimal of 10)

Task No. 02:

"Write an assembly program that prints the name of your favorite book, quote, couplet, movie, drama, color and dish."

Solution:

```
.data
book : .asciiz "Favorite book: A song of ice and fire "
quote: .asciiz "Favorite quote: Take your meals as medicine or take your medicine as meal one day."
couplet: .asciiz "Favorite couplet: \n Nature's first green is gold, Her hardest hue to hold"
movie: .asciiz "Favorite movie: Game of thrones"
drama: .asciiz "Favorite drama: Breaking bad"
color: .asciiz "Favorite color: Blue"
dish: .asciiz "Favorite dish: Biryani"
newline: .asciiz "\n"
.text
main:
li $v0 4
la $a0 book
syscall

li $v0 4
la $a0 newline
syscall

li $v0 4
la $a0, quote
syscall

li $v0 4
la $a0 newline
syscall

li $v0, 4
la $a0 couplet
syscall

li $v0 4
la $a0 newline
syscall

li $v0, 4
la $a0, movie
syscall
```

```
li $v0 4
la $a0 newline
syscall

li $v0, 4
la $a0 drama
syscall

li $v0 4
la $a0 newline
syscall

li $v0, 4
la $a0,color
syscall

li $v0 4
la $a0 newline
syscall

li $v0 4
la $a0 dish
syscall

li $v0 4
la $a0 newline
syscall

li $v0, 10
syscall
```

Output:

```
Favorite book: A song of ice and fire
Favorite quote: Take your meals as medicine or take your medicine as meal one day.
Favorite couplet:
 Nature's first green is gold, Her hardest hue to hold
Favorite movie: Game of thrones
Favorite drama: Breaking bad
Favorite color: Blue
Favorite dish: Biryani

-- program is finished running --
```

Task No. 03:

"Write an assembly program that takes the student information like name, enrollment, registration number, program, semester and section from user and then print it in proper format for ID Card."

Solution:

```
.data
askName: .asciiz "Enter Name: "
name: .space 20
askEnrollment: .asciiz "Enter Enrollment number: "
enrollment:  .space 40
askRegistration: .asciiz "Enter Registration number: "
registration: .space 20
askProgram: .asciiz "Enter Program: "
program: .space 20
askSemester: .asciiz "Enter Semester: "
semester: .word 0
askSection: .asciiz "Enter Section: "
section: .byte 0
dash : .asciiz "--------------------------------------------------"
newline : .asciiz "\n"
nameee: .asciiz "Name: "
enrollmenttt: .asciiz "Enrollment: "
registrationnn: .asciiz "Registration: "
programmm: .asciiz "Program: "
semesterrr: .asciiz "Semester: "
sectionnn: .asciiz "Section: "
```

```
.text
main:
   li $v0, 4
   la $a0, askName
   syscall
   li $v0, 8
   la $a0, name
   li $a1, 20
   syscall
   li $v0, 4
   la $a0, askEnrollment
   syscall
   li $v0, 8
   la $a0, enrollment
   li $a1, 40
   syscall
   li $v0, 4
   la $a0, askRegistration
   syscall
   li $v0, 8
   la $a0, registration
   li $a1, 20
   syscall
   li $v0, 4
   la $a0, askProgram
   syscall
```

```
li $v0, 4
    la $a0, askSemester
    syscall
    li $v0, 5
    syscall
    sw $v0, semester
    li $v0, 4
    la $a0, askSection
    syscall
    li $v0, 12
    syscall
    sb $v0, section
    li $v0, 4
    la $a0, newline
    syscall
    li $v0, 4
    la $a0, dash
    syscall
    li $v0, 4
    la $a0, newline
    syscall
    li $v0, 4
    la $a0, nameee
    syscall
    li $v0, 4
    la $a0, name
    syscall
```

```
 li $v0, 4
la $a0, enrollmenttt
syscall
li $v0, 4
la $a0, enrollment
syscall
li $v0, 4
la $a0, registrationnn
syscall
li $v0, 4
la $a0, registration
syscall
li $v0, 4
la $a0, programmm
syscall
li $v0, 4
la $a0, program
syscall
li $v0, 4
la $a0, semesterrr
syscall
lw $a0, semester
li $v0, 1
syscall
li $v0, 4
la $a0, newline
syscall
```

```
li $v0, 4
la $a0, sectionnn
syscall
lb $a0, section
li $v0, 11
syscall
li $v0, 4
la $a0, newline
syscall
li $v0, 4
la $a0, dash
syscall
li $v0, 10
syscall
```

Output:

```
Enter Name: Abdullah
Enter Enrollment number: 070
Enter Registration number: 95346
Enter Program: BSE
Enter Semester: Fall 2025
Enter Name: Abdullah
Enter Enrollment number: 070
Enter Registration number: 95346
Enter Program: BSE
Enter Semester: 3
Enter Section: B
--------------------------------------------------
Name: Abdullah
Enrollment: 070
Registration: 95346
Program: BSE
Semester: 3
Section: B
--------------------------------------------------
-- program is finished running --
```