# Andrew Souza

# Comp 265 -- Spring 2024

# Introduction to MIPS -- Problem 1

# Please read chapter 2 of the Introduction To MIPS Assembly Language Programming.

# Follow the instructions to download the MARS IDE onto your computer.

# Run your first program using this new IDE by following the steps of Chapter 2.3.

.text # Define the program instruction

main: # Label to define main program

li $v0, 4 # Load 4 into $v0 to indicate a print string

la $a0, greeting # Load the address of the greeting into $a0

syscall # Print greeting. The print is indicated by

# $v0 having a value of 4, and the string

# to print is stored at the address in $a0.

li $v0, 10 # Load a 10 (halt) into $v0

syscall # The program ends

.data # Define the program data

greeting: .asciiz "Hello World" # The string to print



# Andrew Souza

# Comp 265 -- Spring 2024

# Introduction to MIPS -- Problem 2

# Write a program using MIPSzy similator to compute the

# perimeter and area of a rectangle.

# Your program is to get the width and height from user

# input and display the area and perimeter as output to the user.

.text

# Prompt the user to enter the length

li $v0, 4

la $a0, promptLength

syscall

# Read the data from input

li $v0, 5

syscall

move $s0, $v0

# Prompt the user to enter the width

li $v0, 4

la $a0, promptWidth

syscall

# Read the data from input

li $v0, 5

syscall

move $s1, $v0

# Output the length and width

li $v0, 4

la $a0, outputLength

syscall

li $v0, 1

move $a0, $s0

syscall

li $v0, 4

la $a0, outputWidth

syscall

li $v0, 1

move $a0, $s1

syscall

# Process and output perimeter and area

# # Find Perimeter

add $s2, $s0, $s1

mul $s2, $s2, 2

li $v0, 4

la $a0, outputPerimeter

syscall

li $v0, 1

move $a0, $s2

syscall

# # Find Area

mul $s2, $s0, $s1

li $v0, 4

la $a0, outputArea

syscall

li $v0, 1

move $a0, $s2

syscall

# # End Program

ori $v0, $zero, 10

syscall

.data

promptLength: .asciiz "Please enter the length of your rectangle: "

promptWidth: .asciiz "Please enter the width of your rectangle: "

outputLength: .asciiz "Length: "

outputWidth: .asciiz "\nWidth: "

outputPerimeter: .asciiz "\nPerimeter: "

outputArea: .asciiz "\nArea: "

