NAME: OMAR JARALLAH ALGHAMDI ID: 201855000 COURSE/ SECTION: ICS233/ 53

Lab 09 report: Floating-Point

Objectives

- To understand Floating-Point Number Representation according to IEEE 754 Standard.
- To understand MIPS Floating-Point Unit.
- To implement programs using the MIPS Floating-Point Instructions.
- To implement functions that have floating-point parameters and return floating-point results

Introduction

In MIPS, there are multiple data types that can represent the data, such as integer, float, double, char, string. In this lab we are interested in float and double data types. Developers usually prefer using floating-point numbers especially in scientific programs because the calculations should be considerably accurate with minimum errors to prevent false results that may cause severe consequences. MIPS uses compressor-1 to store and preform floating-point operations. There are arithmetic operations, Boolean operations and addresses operations.

Tasks

Task1:

1. **Requirement:** it is required to implement a MIPS program that prompts to enter double precision floating-point numbers x and y and calculates the result according to the given formula:

$$result = \begin{cases} 3.14\left(\sqrt{-\frac{x}{y}}\right), & \frac{x}{y} < 0 \\ \sqrt{2 \times \frac{x}{y}}, & \frac{x}{y} > 0 \end{cases}$$

2. **Approach:** to implement this program it is required to prompt enter number x then prompt him to enter y in double precision format. After the user has entered x and y the program will save these values for later use. Then the program will check if the division quotient is less than zero to determine if the number inside the square root is positive or negative. If the number is negative the program will find the result using this formula ($3.14\left(\sqrt{-\frac{x}{y}}\right)$). at first it will take the negative of the division quotient using negation instruction. Then the program will take the square root of the negated number. After it takes the negation of the division quotient, it will be multiplied by 3.14 that is declared in the data segment (it will load the number then converts it to double precision floating-point number). Then it will print the result then it will be terminated. However, if the division quotient is more than zero, the program will calculate the result using this formula ($\sqrt{2 \times \frac{x}{y}}$)**.** It will multiply the division quotient by two then takes the square root of it. After it takes the square root, it will print the result then it will be terminated.

Tasks

Task2:

1. **Requirement:** it is required to implement a MIPS program that prompts to enter the quiz grades of 12 students then calculates their average.

2. **Approach:** to implement this program it is required to declare the number of students in the program then convert it to single precision floating-point number to calculate the accurate average (number of students should be 12 according to the task requirement). Then it will implement this formula to find the average:

$$\boldsymbol{Averege} = \frac{\sum_{x=grade}^{\boldsymbol{Number\ of\ Students}}(\boldsymbol{x})}{\boldsymbol{Number\ of\ students}}$$

   To implement this formula the first thing it is required to find the sum of the grades. To find the sum of the grades we will declare a loop to add the grades in each iteration. First thing to do in the loop is to check if the number of iterations is less than the number of students. Then the program will prompt the user to enter a grade in single precision floating-point format. Then the program will add the grade to the sum of the grades. The program will iterate until the loop key reaches the number of the students. After the sum has just been calculated, the program will divide it by the number of students then print it. Finally, the program will be terminated.

Conclusion

In conclusion, MIPS offers float-point operations which increase the accuracy of the calculations such as finding the average and the square root of a number.