Assignment 8

Recursive Functions

Introduction

This assignment will implement a recursive function. Because a recursive function calls itself, it must be consider to be a non-leaf function. Therefore, this implementation is no different than implementing a non-leaf function.

This program will create a function named power that will raise a double-precision number to an integer exponent. The exponent may be positive or negative. If it is negative, then the function should use the fact that

$$a^n = \left(\frac{1}{a}\right)^{-n}$$

In other words, replace a with $1/a$ and replace n with $-n$ and then proceed with the same code that is used when the exponent is positive.

The Recursive Algorithm

The algorithm uses repeated squaring. For example, to raise a to the 16th power, we could square four times:

$$a^{16} = \left(\left(\left(a^2\right)^2\right)^2\right)^2.$$

Is a bit more complicated when the exponent is not a power of 2. For example, what if the exponent were 41? Then we use the fact that

$$a^{41} = \left(a^{20}\right)^2 \times a,$$

which reduces the problem to finding a^{20}:

$$a^{20} = \left(a^{10}\right)^2.$$

Now we need to find a^{10}:

$$\begin{aligned} a^{10} &= \left(a^5\right)^2, \\ a^5 &= \left(a^2\right)^2 \times a, \\ a^2 &= a^2. \end{aligned}$$

Note that whenever the exponent is even, we simply square the value of a raised to half the exponent, and when the exponent is odd, we do the same (with the truncated half of the exponent) and then multiply by a.

I am providing to you the C++ program `power_func.cpp` that implements this algorithm in C. You should rewrite it in unstructured C and then write it in MIPS.

Note that the base is a `double`. That means that you will need to use the floating-point processor in your program. Recall that the registers `$f12-13` are used to pass double-precision arguments and registers `$f0-f1` are used to return double-precision values. Furthermore, the MIPS convention is that registers `$f20-f31` are preserved over function calls, while the others are not. So, treat these registers as you would treat the `$s` registers and use the others as you would use the `$t` registers.

Due Date

Place your work in a folder named `Assignment 8` and drop it in the dropbox. It is due by midnight Wednesday, October 2.