Cosc 2P12
Assignment 4

(Due date for assignment is Thursday December 3rd,4:00 p.m. est., Late date Monday December. 7th, 4:00 p.m.est)

Part A:

Below is a variation of a Fibonacci sequence, not an original. Many such variations exist. Implement the below function using the MARS Mips assembler. Call the function repeatedly to produce the output as shown below. For your convenience you are given the function in symbolic and C code.

ZIBONACCI

This is a weird rendition of Fibonacci which tends to calculate results which appear to zig and zag.

zib(0) = 1
zib(1) = 1
zib(2) = 2
zib(2n+1) = zib(n) + zib(n-1) + 1, if n>0 (odd values 3 and higher)
zib(2n) = zib(n) + zib(n+1) + 1, if n>1 (even values 4 and higher)
The C program to calculate this looks like:

```
1
2     int zib(int n) {
3        if (n == 0)
4           return 1;
5        else if (n == 1)
6           return 1;
7        else if (n == 2)
8           return 2;
9        else if (n%2 == 1 && n >= 3) // odd
10          return zib((n-1)/2) + zib((n-1)/2-1) + 1;
11       else if (n%2 == 0 && n >= 4) // even
12          return zib(n/2) + zib(n/2+1) + 1;
     }
```

Here is the output for 1 to 20:

1 2 3 6 4 10 6 11 10 15 11 17 15 18 17 22 18 26 22 27

Each recursive call will need an activation record. These should be created and destroyed using the conventions set out in lecture. Following convention is important to show your understanding of activation records.

Be sure to properly document your code.

Part B

Written portion: Using your student number which will appear in the following format, abcDEFG. This will produce the two base 10 numbers, D.E x 10^2, and F.G x 10^3. Take your student number and substitute in the digits as shown above. Using the FPS website (see course home page) convert each over to a normalized binary form. Each should look something like 1.xxxxxxxx * 2^y. Depending on your student number, the number of significant digits will vary.

For those students who have DE and/or FG equal to zero, i.e. $0.0x10^2$, then substitute 34 for any occurrence of 00 to ensure a non-zero term.

Using the FPS multiplication shown in class, multiply the above numbers together, producing a FPS result. Show all work, that is where the marks are. You can verify your work by doing a base 10 multiplication and using the FPS website to verify the binary result.

Submission

This submission will be submitted electronically as a MIPS assembly file and a written portion.

The TA will be running your program to ensure it is fully functional. Make the marker happy!!!

For the electronic submission, use Sakai, an assignment 3 submission will be available.

The End