Hiren Patel

Readme.pdf

Mystery

It is very clear that this program was Fibonacci series producing the value of the nth term. This program was also quite inefficient. This program used recursion to complete the task, however, it called back twice (fib(n-1) + fib(n-2)). This requires more and more time as you go higher and higher and the time complexity clearly suffers greatly. A much more efficient way would've been:

```
Static int first = 0;

Static int second = 1;

Static int sum;

If (input > 0)

{

        Sum = first + second;

        First = second;

        Second = sum;

        Fibonacci(input -1);

}

If (input == 0)

        Return sum;
```

This version of the Fibonacci series only requires one call back (Fibonacci(input -1) ) and this greatly improves the time complexity. This version however requires the input value be subtracted by one and have a switch statement for 0 and -1 specifically, but this version is indeed much more efficient.

I went about the figuring out mystery step by step. I noticed that it recursively called itself twice and then implemented the add function(which I don't know why was added). I thought it might be Fibonacci by the way it was implemented. I tested out a few numbers and it matched the Fibonacci sequence.

As for changes in the code in optimized vs unoptimized (o vs u), the o code had my add function essentially reduced to a single line using leal. It encapsulated all the function essentially into one with the necessary parts already in the registers. In the dothething function, there was a lot less moving in the o code vs the u code. The o code implemented only 7 move statements whereas the u code had 15 move statements. Also, leal was used in the o code but not the u code. The conditional statements and jumps stayed the same though. In the main, there was again a lot less move statements as opposed to the u code. The o code also didn't implement atoi but did implement strtol, which converts a string into a long integer.

The o code seemed to have much less move statements and also implemented more leal statements. It also recognized not to use atoi and instead opt for strtol.