

Recursive vs iterative

For this test I was curious to see how well Udon executes recursive methods, in the previous test I already compared the execution time of recursive functions in U# and C#, but here I wanted to compare the execution time between a recursive function and an iterative function

I executed two methods `Benchmark1` and `Benchmark2` and compared their execution time.

```
[RecursiveMethod]
private int Recursive(int n)
{
    if (n <= 0)
        return 0;
    else
        return 1 + Recursive(n - 1);
}

private int NotRecursive(int n)
{
    int ret = 0;
    while (n > 0)
    {
        ret++;
        n--;
    }
    return ret;
}

public override void Benchmark1()
{
    Recursive(50000);
}

public override void Benchmark2()
{
    NotRecursive(50000);
}
```

Benchmark1 calculates a value recursively, Benchmark2 calculates it iteratively.

B1 : 532.552 ms (6.6 times slower)

B2 : 79.8591 ms

Recursive functions are much slower, they should be avoided!

The reason is that Udon builds a custom stack to save the variables of each function call, otherwise the next function call would override the variables from the previous function call. Implementing such a stack is performance heavy.

Revision #2

Created 15 February 2023 19:49:25 by MyroP

Updated 16 February 2023 20:44:46 by MyroP