

实验结果复现:

test.py

[illegible]

test.cpp

[illegible]

发现使用c++写的程序会比使用python写的程序快

Q1: 为什么使用c++写的程序会比使用python 写的程序快?

A1: 首先, C++是一种编译型语言, 编译器将 C++代码转换成机器语言, 使得最终生成的可执行程序在机器上更快; 而 Python 是一种解释型语言, 在运行的过程中, 需要编译+解释, 导致了更多的时间消耗。其次, C++是一种静态类型语言, 静态类型语言在编译时就能对代码进行优化, 使代码的执行效率更高; 而 Python 是一种动态类型语言, 在运行时需要动态地进行类型推断和管理, 这会导致一定的性能损耗。因此, 使用 c++写的程序会比使用python 写的程序快。

Q2: 去掉"test.cpp"中的#include<gmpxx.h>, 并将 mpz_class 改成 int 后, 计算 100 的阶层的运行结果如何? 为什么会出现这样的结果?

A2: 运行结果输出为 0。int 类型的范围是[-2147483648, 2147483647]。而计算 100 的阶层的值过大，导致数值溢出。当递归进行至 f (33) 时，返回值为-2147483648。

而-2147483648 乘上接下来的偶数 34 后，数值依旧溢出，末尾被截断后返回 0 值。因

此接下来的递归中均为 $n*0$ ，最终程序运行结果为 0。

```
root@horizon22:/mnt/hgfs/share# time ./test2
0
real    0m0.003s
user    0m0.002s
sys     0m0.000s
```

Q3: 除了编程语言外，影响程序运行快慢还可能有哪些因素？

A3:

因素一：不同的算法的时间复杂度不同，导致程序的运行快慢会不同。

因素二：硬件的差异会影响程序运行的快慢。硬件性能越好一般会让程序运行越快。

因素三：代码质量也会影响快慢。比如过多循环嵌套和冗余代码会降低性能。