总结

上午程设A课堂上，完成了project0的程序，那时候1s只能跑100左右，困难主要在进制转换和int只用在二进制单位运算上面，耗时也主要分布在上面，打算下午优化。

```
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ time ./a

real    0m0.865s
user    0m0.864s
sys     0m0.000s
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ diff pi.out sta
ndard10K.out
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$
```

```
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ time ./a

real    1m28.490s
user    1m28.400s
sys     0m0.036s
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ diff pi.out sta
ndard100K.out
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$
```

以上是我project1跑分结果，看样子和柱爷还差得远，下午我采用了压位、开O2编译等优化手段，但由于时间关系，没来得及进一步优化，先存档。

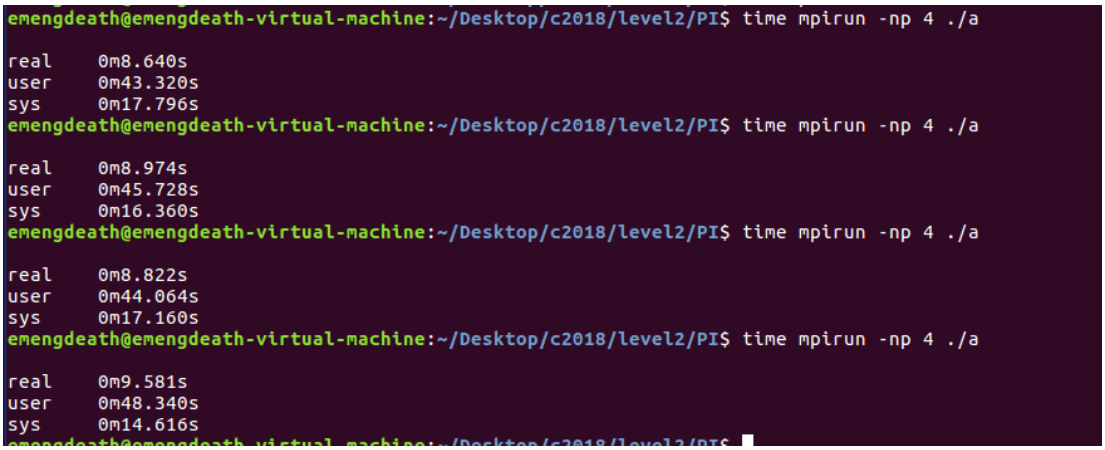
```
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ time ./a

real    0m0.529s
user    0m0.528s
sys     0m0.000s
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ diff pi.out standard10K.out
```

```
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ time ./a

real    0m52.995s
user    0m52.860s
sys     0m0.044s
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ diff p
pi.out    pi.out~   project0/ project1/ project2/
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ diff pi.out standard100K.out
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$
```

以上为晚上进行成果，对于串行代码热点优化，主要在于将求和部分合并了，以及用位运算代替了取模运算等，首次突破100K数据1min。后面比拼并行计算能力。希望能赶上柱爷成绩～～～

```
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ time mpirun -np
 4 ./a

real    0m18.652s
user    1m44.444s
sys     0m1.632s
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$ diff pi.out sta
ndard100K.out
emengdeath@emengdeath-virtual-machine:~/Desktop/c2018/level2/PI$
```

次日早上，多线程和多进程优化100K数据进入20s，热点变为最后进制转换的多次2的除法里面，希望下午能完成，匆忙记一波。



下午终于100K数据优化进了10s，第一次跑了1M数据，差点以为跑不出来，结果跑了800+s，10K数据由于虚拟机不稳定，就算是打平了。总的说，算是破了柱爷记录了，由于我用多进程，所以time.h的clock计时似乎不太准，我就截图了命令行time计时，以及内部采用MPI自带的计时器（精确到1e-6，不过少了一开始开进程的时间和输出pi的时间，不过柱爷也没算输出时间，开进程时间不多0.2s左右），现在为止都是人脑优化，后面打算下些辅助的工具来继续优化，看了柱爷代码发现比较简洁，采用高精度模板，单靠多线程就10s左右了，厉害啊，而我为了仅仅适应这个工作，强行源码优化了不少地方，还需努力啊！！