

高精度加减乘

这个程序可以实现加减乘运算只有一次的表达式，支持高精度，不支持括号。

加减法的时间复杂度为 $O(n)$ ；乘法计算本质上是一个多项式乘法，所以使用了FFT快速傅立叶变换算法，把乘法的时间复杂度降低到了 $O(n \log n)$ ，一秒内可以实现位数为 10^6 的两个十进制数相乘。

代码见 `calculator` 文件夹，以下是一些运算结果：

```
● root@730aa3418e90:/ws/calculator/build# ./main
-99999.09879*124809.00
-2480787520.88111
● root@730aa3418e90:/ws/calculator/build# ./main
13459871394571934571934756198347+23467820340134586018934560136450183465134650374787896978435
23467820340134586018934560149910054859706584946722653176782
● root@730aa3418e90:/ws/calculator/build# ./main
34952876-1345876103456012873465013894635081364501345
-1345876103456012873465013894635081329548469
● root@730aa3418e90:/ws/calculator/build# ./main
4350872350295328457208670283460293846023984610398467139845670139468.2392389787652359578*134508374
504737134751571239560123571239582389512893561093256013598712873561829561.134687523897456103874651
5852287675158298943329654830898372809668623342874140000022768898300138766246430116595979585351769
71451373984799106012033765266209760122284402083071.7536129928748405970698815559786528491257278
```

表达式求值

表达式求值的思想很简单了，基本上就是暴力写大工程。代码见 `expression` 文件夹。以下是一些运算结果：

```
● root@730aa3418e90:/ws/expression/build# ./main
123-21*(sin(2.0)+sqrt(16))+5/(1+3.2)
18.714278
● root@730aa3418e90:/ws/expression/build# ./main
(92-23)*3+45/5
216.000000
● root@730aa3418e90:/ws/expression/build# ./main
(2.34+5.67)*3.5/2-1.23+4.56/2*(8.9-3.4)
12.372955
```