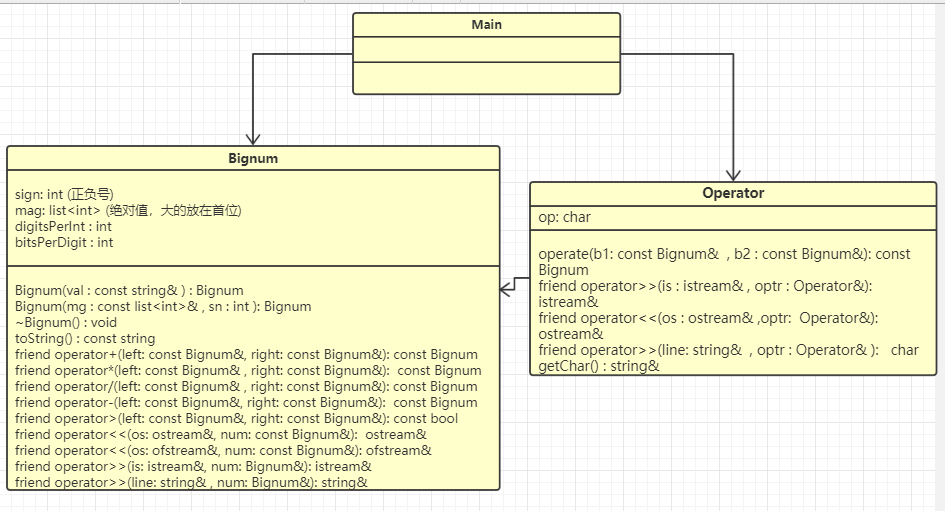
# Lab4 说明文档

## 类设计

Main这个类负责读取文件， Bignum 负责存储和计算结果， Operator 存储运算符并调用相应的加减乘除操作。



Bignum 输入： 可以通过 cin >> num 初始化，或者 string >> num 方式，不提供 ifstream >> num 因为可能会出现问题。

输出： 可以通过 cout << num 输出，或者 ofstream << num 输出。

## 算法实现

1. 存储： 使用sign(int) 存储正负号， 使用mag (list<int>) 存储绝对值，大的数字放在前面，每个int存储9位10进制数字，相当于将10进制的数字做了一个切割，放入相应的数组中。（例如 123456789123456789 将变为123456789 | 123456789 存储）
2. 加减法： 首先，判断符号，然后进行绝对值的运算。 加法： 超过了10^9-1，向前进一位，从低位到高位进行加法运算；减法： 数字不够减了，向高位借一个1，从低位到高位进行减法运算。（例如， 999999999 + 1 = 1 | 000000000）
3. 乘法： 两个数字各个位数相乘（以九个数字为一个位），然后相加，设第一个数字的位数为n，第二个为m，那么复杂度为O（n\*m）
4. 除法： 逐位进行删除（例如， 987 / 12 , 先对987的前面两位进行操作，98/12 = 8 , 然后剩下 98 – 12\* 8 = 2， 27 ， 进行 27 / 12 = 2, 得到82）, 设第一个数有n位，第二个数有m位， 则复杂度为O（n-m），但实际上常数会很大，主要在于每次逐个相除的代价比较高。（当然已经用了一些方法优化了。）

## 使用说明

1. 在windows命令行下，输入Bignum.exe sample.in sample.out

将会读取并计算相关的文件。如果不按照这个格式输入，会输出提示。

1. 在完成文件的计算和输出后，还可以进行从命令行输入公式进行计算。

首先，输入compute。然后输入计算数的个数，接着，就可以输入计算式子了。

注意： 格式，数字和符号之间要有空格。