

高精度除法（高精度除以低精度）

示例：

尝试计算 10112321321321321321321312 除以 2

解析

高精度除以低精度是最为简单的除法，回忆一下我们小学数学学的除法是如何计算的？对，我们只要从最高位开始进行不断的试除即可，如果当前的数字不够整除除数，则需要向下进行借位，编码如下。

编码

```
#include<bits/stdc++.h>

using namespace std;
//被除数
string divi;
//除数和余数
long long b, d;
//被除数
int a[10001];
//最终的结果
int c[10001];

int main() {
    //读入被除数和除数
    cin >> divi >> b;
    //把被除数的位数存在len里
    int len = divi.length();
    //将被除数转化为数字进行存储
    for (int i = 1; i <= len; i++) {
        a[i] = divi[i - 1] - '0';
    }
    //循环试除，如果值得大小不够则向下借位
    for (int i = 1; i <= len; ++i) {
        c[i] = (d * 10 + a[i]) / b;
        d = (d * 10 + a[i]) % b;
    }
    int lenc = 1;
    //整理位数，删除前导0
    while (c[lenc] == 0 && lenc < len) {
        lenc++;
    }
    //输出结果
```

```
for (int i = lenc; i <= len; ++i) {  
    cout << c[i];  
}  
return 0;  
}
```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

