基础算法 II

高精度算法

- ① A + B　10^6（位数）
- ② A - B　10^6（位数）
- ③ A × a　$len(A) \le 10^6$　$a \le 10^9$
- ④ A ÷ a

1、如何存储

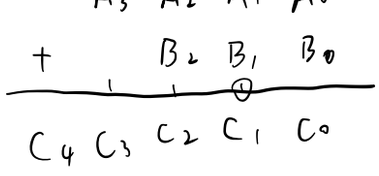
① 将大整数存在数组

② 低位存在 0，高位存在 n

↳ 方便进位

2、如何运算？

模拟人工加法（/）



三个数相加 A，B，T 进位

```cpp
#include <iostream>
#include <cstring>
#include <algorithm>
#include <vector>

using namespace std;

const int N = 100010;

vector<int> add(vector<int>& A,vector<int>& B)
{
    int t = 0;
    vector<int> C;
    for(int i=0;i<A.size() || i < B.size();i++)
    {
        if(i<A.size()) t+=A[i];
        if(i<B.size()) t+=B[i];
        C.push_back(t % 10);
        t /= 10;
    }
    if(t) C.push_back(1);
    return C;
}

int main()
{
    vector<int> A,B;
    string a,b;
    cin>>a>>b;
    for(int i=a.size()-1;i>=0;i--)
    {
        A.push_back(a[i]-'0');
    }

    for(int i=b.size()-1;i>=0;i--)
    {
        B.push_back(b[i]-'0');
    }
    auto C = add(A,B);
    for(int i=C.size()-1;i>=0;i--) printf("%d",C[i]);
    return 0;
}
```

高精减法

$A_1\ A_2\ A_3\ A_4$

$-\quad B_1\ B_2\ B_3$

是否要借位

$A_i - B_i - t \begin{cases} \geq 0 & A_i - B_i - t \\ < 0 & A_i - B_i + 10 - t \end{cases}$

假定：$A \geq B$

若 $A < B \rightarrow B \rightarrow A \rightarrow$ 加负号 $(-(B-A))$

a. 如果有负数？

判断借位符号，打标记

两个数相减，一定可以转化成，打符号标记，分类讨论

$a-b \begin{cases} |a|-|b| \\ |a|+|b| \end{cases}$

a正，b正 → normal $|a|-|b|$

a负，b负 → $a-b = -|a|+|b| = |b|-|a|$

a正，b负 → $a-b = |a|+|b|$

a负，b正 → $a-b = -|a|-|b| = -(|a|+|b|)$

```cpp
#include <iostream>
#include <cstring>
#include <algorithm>
#include <vector>

using namespace std;

const int N = 100010;

// A>=B
bool cmp(vector<int>& A,vector<int>& B)
{
    if(A.size() != B.size()) return A.size()>B.size();
    else{
        for (int i = A.size()-1; i >=0; i -- )
        {
            if(A[i]!=B[i]) return A[i]>B[i];
        }
        return true;
    }
}

vector<int> sub(vector<int>& A,vector<int>& B)
{
    int t = 0;
    vector<int> C;
    for(int i=0;i<A.size();i++)
    {
      t = A[i] - t;
      if(i<B.size()) t-=B[i];
      C.push_back((t+10)%10);
      if(t<0) t=1;
      else t=0;
    }
    while(C.size()>1&&C.back()==0) C.pop_back();
    return C;
}

int main()
{
    vector<int> A,B;
    string a,b;
    cin>>a>>b;
    for(int i=a.size()-1;i>=0;i--)
    {
        A.push_back(a[i]-'0');
    }

    for(int i=b.size()-1;i>=0;i--)
    {
        B.push_back(b[i]-'0');
    }
    if (cmp(A,B))
    {
        auto C = sub(A,B);
        for(int i=C.size()-1;i>=0;i--) printf("%d",C[i]);
    }
   else{
       auto C = sub(B,A);
       printf("-");
       for(int i=C.size()-1;i>=0;i--) printf("%d",C[i]);
   }
    return 0;
}
```

高精度乘法

$A_5\ A_4\ A_3\ A_2\ A_1\ A_0$

$\times$ a

个位：$A_0 \times b \ \%\ 10$

进位：$A_0 \times b \ / \ 10$

进位：$A_1 \times b + t_1$ 进位

实例：A 1 2 3

B × 1 2

$C_3\ C_2\ C_1\ C_0$

$C_0 = 3\times12 \ \%\ 10 = 6 \quad t_1 = 3\times12/10$

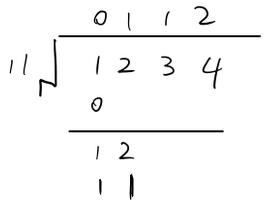
$C_1 = (2\times12 + t_1) \ \%\ 10 = 7 \quad t_2 = (2\times12 + t_1)/10 = 2$

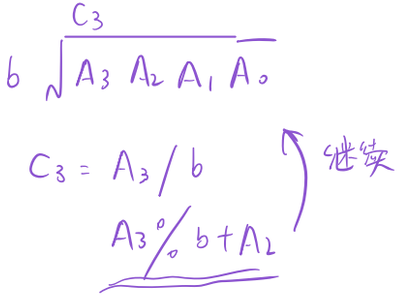
$C_3 = (1\times12 + t_2) \ \%\ 10 = 4 \quad t_3 = (1\times12 + t_2)/10 = 1$

$C_4 = 1$

```
#include <iostream>
#include <cstring>
#include <algorithm>
#include <vector>

using namespace std;

vector<int> mul(vector<int>& A,int b)
{
    int t = 0;
    vector<int> C;
    for(int i=0;i<A.size() || t ;i++)
    {
        if(i < A.size()) t += (A[i] *b);
        C.push_back(t % 10);
        t /= 10;
    }
    return C;
}

int main()
{
    vector<int> A;
    string a;
    int b;
    cin>>a>>b;
    if(b==0){ printf("%d",0);return 0;}
    for(int i=a.size()-1;i>=0;i--)
    {
        A.push_back(a[i]-'0');
    }

    auto C = mul(A,b);
    for(int i=C.size()-1;i>=0;i--)
printf("%d",C[i]);

    return 0;
}
```

高精度除法






```cpp
#include <iostream>
#include <cstring>
#include <algorithm>
#include <vector>

using namespace std;

vector<int> div(vector<int>& A,int b,int &r)
{
    vector<int> C;
    for(int i=A.size()-1;i>=0;i--)
    {
        r = r*10 + A[i];
        C.push_back(r/b);
        r %= b;
    }
    reverse(C.begin(),C.end());
    while(C.size()>1 && C.back()==0) C.pop_back();
    return C;
}

int main()
{
    vector<int> A;
    string a;
    int b,r=0;
    cin>>a>>b;
    if(b==0){ printf("%d",0);return 0;}
    for(int i=a.size()-1;i>=0;i--)
    {
        A.push_back(a[i]-'0');
    }

    auto C = div(A,b,r);
    for(int i=C.size()-1;i>=0;i--) printf("%d",C[i]);
    printf("\n%d",r);

    return 0;
}
```