



## project2分享



主讲人 徐福来



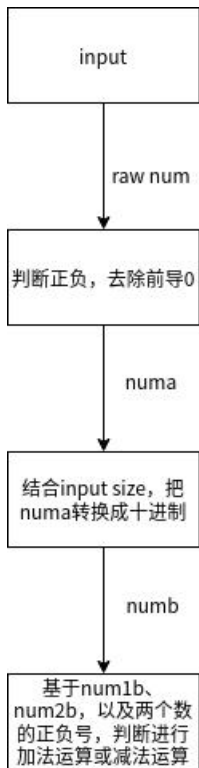
## ➤ 第一部分：思路概述

本题可以拆分成大数加减法和任意进制转换两个自问题求解

## ➤ 第二部分：关键模块介绍

## ➤ 第三部分：代码分享

# 思路概述



预处理, 将main函数传参和调用的txt文件转换成两个用string保存的大数raw\_num1和raw\_num2, input\_size, output\_size

对num1和num2分别判断正负, 去除前导0, 用numa存储数字, 用一个bool存储正负号  
例如: 将-000000123456转换成string numa = 123456, bool is\_positive = false

利用长除法, 将两个input\_size的numa分别转换成十进制的numb

考虑正负号和两个numb的size, 决定输出的正负号和选择的运算符  
进行大数加减法运算

对最终计算结果, 结合output\_size和正负号, 输出

# 纲要

---

- 第一部分：思路概述
- 第二部分：关键模块介绍
- 第三部分：代码分享

# 基础转换函数

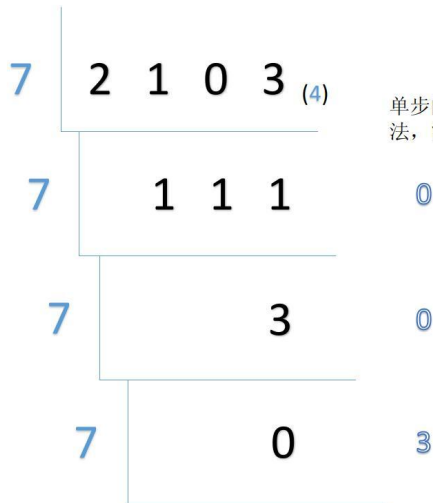
因为大数是以string形式存储的，  
所以需要“数字”概念和“字符”概念  
的相互转换函数

数字----->字符: TransN2C

字符----->数字: TransC2N

```
32 int TransC2N(const char raw, const int limit) {
33     int raw_num = 0;
34     if (raw >= '0' && raw <= '9') {
35         raw_num = raw - '0';
36     } else if (raw >= 'A' && raw <= 'Z') {
37         raw_num = raw - 'A' + 10;
38     } else if (raw >= 'a' && raw <= 'z') {
39         raw_num = raw - 'a' + 10;
40     }
41     if (raw_num >= limit) {
42         cout << "trans one num error, ignore print below" << endl;
43         return 100;
44     }
45     return raw_num;
46 }
47
48 char TransN2C(const int raw, const int limit) {
49     if (raw >= 0 && raw <= 9) {
50         return '0' + raw;
51     } else if (raw >= 10 && raw <= 35) {
52         return 'a' + raw - 10;
53     }
54     return '!';
55 }
```

# 长除法的思路 and 代码



单步的运算需要自定义除法，能够得到整除数和余数



获得结果300<sub>(7)</sub>

```
56 // n进制的str除以m, 余数是remain
57 string division(string str, int m, int n, int & remain){
58     int a;
59     remain = 0;
60
61     for(int i = 0; i < str.size(); i++){
62         int tmp_n = TransC2N(str[i], n);
63         a = (n * remain + tmp_n);
64         if (tmp_n == 100) {
65             return "?";
66         }
67         str[i] = TransN2C(a / m, n);
68         remain = a % m;
69     }
70     //去掉多余的0 比如10/2=05
71     int pos = 0;
72     while(str[pos] == '0'){
73         pos++;
74     }
75     return str.substr(pos);
76 }
77
78
79 // n进制的str化成m进制
80 string conversion(string str, int m, int n){
81     string result = "";
82     int a;
83     //因为去掉了多余的0, 所以终止条件是字符串为空 例: 当上一步运算结果为"0"时, 实际上返回的结果为""
84     while(str.size() != 0){
85         str = division(str, m, n, a);
86         result = TransN2C(a, m) + result;
87         if (str == "?") {
88             return "?";
89         }
90     }
91     if (result.front() == '0') {
92         int i = 0;
93         while (result[i] == '0') {
94             i++;
95         }
96         result = result.substr(i);
97     }
98     return result;
99 }
```

# 运算符控制



同号用加法，异号用减法，记录正负，控制s1比s2长以便于补0

```
160 // 保证s1比s2长
161 string MyFun(string s1, string s2, bool is_num1_positive, bool is_num2_positive, bool& is_res_positive) {
162     bool is_add = true;
163     if (is_num1_positive && is_num2_positive) {
164         is_res_positive = true;
165         return MyAdd(s1, s2);
166     } else if ((!is_num1_positive) && (!is_num2_positive)) {
167         is_res_positive = false;
168         return MyAdd(s1, s2);
169     } else if (is_num1_positive && (!is_num2_positive)) {
170         is_res_positive = true;
171         return MySubtract(s1, s2);
172     } else if ((!is_num1_positive) && is_num2_positive) {
173         is_res_positive = false;
174         return MySubtract(s1, s2);
175     }
176     return "?";
177 }
```

# 大数加减法实现

为了方便计算，翻转string，从而便于从低位向高位运算；

对size较小的s2，进行高位补0

在输出时再次翻转便于阅读

```
string MyAdd(string s1, string s2) {
    string res = "";
    reverse(s1.begin(), s1.end());
    reverse(s2.begin(), s2.end());
    int length1 = s1.size();
    int length2 = s2.size();
    for (int i = 0; i < (length1 - length2); i++) {
        s2.push_back('0');
    }
    int plus = 0;
    for (int i = 0; i < s1.size(); i++) {
        int cur = s1[i] - '0' + s2[i] - '0' + plus;
        if (cur >= 10) {
            cur = cur - 10;
            plus = 1;
        } else {
            plus = 0;
        }
        char tmp_c = TransN2C(cur, 10);
        res.push_back(tmp_c);
    }
    if (plus >= 1) {
        char tmp_c = TransN2C(plus, 10);
        res.push_back(tmp_c);
    }
    reverse(res.begin(), res.end());
    return res;
}
```

```
string MySubtract(string s1, string s2) {
    string res = "";
    reverse(s1.begin(), s1.end());
    reverse(s2.begin(), s2.end());
    int length1 = s1.size();
    int length2 = s2.size();
    for (int i = 0; i < (length1 - length2); i++) {
        s2.push_back('0');
    }
    int plus = 0;
    for (int i = 0; i < s1.size(); i++) {
        int cur_plus = 0;
        int x = s1[i] - s2[i] + plus;
        if (x < 0) {
            x = 10 + x;
            cur_plus = -1;
        } else {
            cur_plus = 0;
        }
        char tmp_c = TransN2C(x, 10);
        res.push_back(tmp_c);
        plus = cur_plus;
    }
    while (res.back() == '0') {
        res.pop_back();
    }
    reverse(res.begin(), res.end());
    return res;
}
```



# 纲要

---

- 第一部分：思路概述
- 第二部分：关键模块介绍
- 第三部分：代码分享

# 获取参数

```
180 int main(int argc, char** argv) {
181     int input_size = 0;
182     int output_size = 0;
183     string raw_data = "";
184     vector<string> str;
185     for (int i = 0; i < argc; i++) {
186         str.push_back(argv[i]);
187     }
188
189     if (argc == 1) {
190         cout << "U don't have raw.txt, bye" << endl;
191         return 0;
192     } else if (argc == 2) {
193         cout << "U don't assign element, set input size 10, output size 10" << '\n';
194         input_size = 10;
195         output_size = 10;
196     } else if (argc == 3) {
197         for (auto x : str[2]) {
198             cout << x - '0' << " ";
199             input_size = input_size * 10 + x - '0';
200         }
201         cout << "U don't assign output size , set output size 10 " << '\n';
202         output_size = 10;
203     } else if (argc == 4) {
204         for (auto x : str[2]) {
205             input_size = input_size * 10 + x - '0';
206         }
207         for (auto x : str[3]) {
208             output_size = output_size * 10 + x - '0';
209         }
210     }
211     raw_data = str[1];
212
213     ifstream infile;
214     infile.open(raw_data);
215     string raw_num1;
216     cout << "Reading from the file" << endl;
217     infile >> raw_num1;
218     cout << "raw_num1 is " << raw_num1 << endl;
219     string raw_num2;
220     infile >> raw_num2;
221     cout << "raw_num2 is " << raw_num2 << endl;
222
223     // get input size
224     cout << "input size is " << input_size << '\n';
225     if (input_size <= 1 || input_size >= 37) {
226         cout << "wrong input size ,bye" << endl;
227         return 0;
228     }
229     // get output size
230     cout << "output size is " << output_size << '\n';
231     if (output_size <= 1 || output_size >= 37) {
232         cout << "wrong output size ,bye" << endl;
233         return 0;
234     }
}
```

# init模块

```
5
6 // 求得正负, 去掉前导零
7 bool InitRawNum(const string raw_num, string& res_num, bool& positive) {
8     if (raw_num.empty()) {
9         return false;
10    }
11    int index = 0;
12    bool is_positive = true;
13    for (; index < raw_num.size(); index++) {
14        if (index == 0) {
15            if (raw_num[index] == '-') {
16                is_positive = false;
17                continue;
18            } else if (raw_num[index] == '+') {
19                continue;
20            }
21        }
22
23        if (raw_num[index] != '0') {
24            break;
25        }
26    }
27    positive = is_positive;
28    res_num = raw_num.substr(index);
29    return true;
30 }
```

# 效果



```
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ls
raw  test2  test2.cpp
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2
U don't have raw.txt, bye
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw
U don't assign element, set input size 10, output size 10
Reading from the file
raw_num1 is 1234
raw_num2 is 23898
input size is 10
output size is 10
num1a is 1234
num2a is 23898
num1b is 1234
num2b is 23898
result in 10 size is 25132
result in output size is : 25132
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 12
1 2 U don't assign output size , set output size 10
Reading from the file
raw_num1 is 1234
raw_num2 is 23898
input size is 12
output size is 10
num1a is 1234
num2a is 23898
num1b is 2056
num2b is 47924
result in 10 size is 49980
result in output size is : 49980
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 12 15
Reading from the file
raw_num1 is 1234
raw_num2 is 23898
input size is 12
output size is 15
num1a is 1234
num2a is 23898
num1b is 2056
num2b is 47924
result in 10 size is 49980
result in output size is : ec20
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 12 10000
Reading from the file
raw_num1 is 1234
raw_num2 is 23898
input size is 12
output size is 10000
wrong output size ,bye
```

基本输入输出测试

```
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 10 10
Reading from the file
raw_num1 is 1234
raw_num2 is -23898
input size is 10
output size is 10
num1a is 1234
num2a is 23898
num1b is 1234
num2b is 23898
result in 10 size is -22664
result in output size is : -22664
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 10 20
Reading from the file
raw_num1 is 1234
raw_num2 is -23898
input size is 10
output size is 20
num1a is 1234
num2a is 23898
num1b is 1234
num2b is 23898
result in 10 size is -22664
result in output size is : -2gd4
```

减法测试

# 效果

```
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 20 10
Reading from the file
raw_num1 is 1234
raw_num2 is BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB
input size is 20
output size is 10
num1a is 1234
num2a is BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB
num1b is 8864
num2b is 31082000168421052631578947368421052631
result in 10 size is 31082000168421052631578947368421061495
result in output size is : 31082000168421052631578947368421061495
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$ ./test2 raw 35 24
Reading from the file
raw_num1 is 1234QWERTYUIOKJHGFDCVBN
raw_num2 is -BBBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB
input size is 35
output size is 24
num1a is 1234QWERTYUIOKJHGFDCVBN
num2a is BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB BBBB
num1b is 9878276315759724586006119327639133
num2b is 194038118257378923735272780279902850880342371
result in 10 size is -194038118247500647419513055693896731552703238
result in output size is : -17h6mh3fdm2mgl2nmh728800h6agi88am
xfl@xfl-Latitude-3410:~/shenlan/福来-project2/code$
```

复杂输入输出测试

Q&A

感谢各位聆听 !  
Thanks for Listening

