**《程序设计基础课程设计》实验报告**

班级: 2班

姓名: 周钰坤,申康

学号: 21009200438

所选题目: 1\_1

第1\_1 题

算法描述: 重点是用数组模拟，加法上主要是实现各个位数的相加，然后之后再实现进位，减法的实现是依靠各位相减，然后在补位，但是缺点是没有实现减法的相加减

源代码:

Process.cpp

#include<iostream>

#include<vector>

using namespace std;

int Compare(string a,string b) //大数比较设定a>b返回1 a<b返回-1 a=b返回0

{

    if(a.size()!=b.size())

    {

        if(a.size()>b.size())

            return 1;

        else

            return -1;

    }

    else

    {

        if(a>b)

            return 1;

        else if(a==b)

            return 0;

        else

            return -1;

    }

}

string Add(string a,string b)           //高精度加法

{

    int n=max(a.size(),b.size())+1;

    vector<int>ans(n,0);

    int i=a.size()-1,j=b.size()-1,k=n-1;

    while(i>=0&&j>=0)

    {

        ans[k--]=(a[i--]-'0')+(b[j--]-'0');

    }

    //检测是否有某个数的高位未加完将其直接存入数组中

    while(j>=0)

    {

        ans[k--]=(b[j--]-'0');

    }

    while(i>=0)

    {

        ans[k--]=(a[i--]-'0');

    }

    string c="";//创建一个字符串去存储答案

    for(int i=n-1;i>0;i--)

    {

        if(ans[i]>=10)

        {

            ans[i]-=10;

            ans[i-1]++;

        }

        c.insert(0,1,ans[i]+'0');//处理后的结果转化为字符插入结果字符的首位

    }

    if(ans[0]>0)

    {

        c.insert(0,1,ans[0]+'0');

    }

    return c;//返回答案

}

string Subtract(string a,string b)//高精度减法(整体思想同加法)

{

    string c="";               //创建一个字符串去存储答案

    if(Compare(a,b)==0)

        return "0";

    if(Compare(a,b)==-1)

    {

        swap(a,b);

        c.push\_back('-');

    }

    int n=a.size();

    vector<int>ans(n,0);

    int i=a.size()-1,j=b.size()-1,k=n-1;

    int t=0;

    while(i>=0)

    {

        char nowb;

        if(j>=0) nowb=b[j];

        else nowb='0';

        ans[k]=a[i]-nowb-t;

        if(ans[k]<0)

        {

            t=1;

            ans[k]+=10;

        }

        else t=0;

        k--,i--,j--;

    }

    bool flag=true;

    for(int i=0;i<n;i++)

    {

        if(flag&&ans[i]==0)

            continue;

        flag=false;        //一旦存入数更改标志位

        c.push\_back(ans[i]+'0');

    }

    return c;              //返回结果

}

int main()

{

    string a,b;

    string add\_ans,subtract\_ans,multiply\_ans,factorial\_ans;

    vector<string>divide\_ans;

    int compare\_ans;

    cin>>a>>b;

    compare\_ans=Compare(a,b);

    cout<<compare\_ans<<endl;

    add\_ans=Add(a,b);

    cout<<add\_ans<<endl;

    subtract\_ans=Subtract(a,b);

    cout<<subtract\_ans<<endl;

    cout<<divide\_ans[0]<<endl<<divide\_ans[1]<<endl;

    return 0;

}

测试程序:

编译: g++ process.cpp -o process.exe

