INFIX TO POSTFIX

#include <cstdio>

#include <iostream>

#define sz(a) ((int)a.size())

using namespace std;

const int MAXN=1e6;

const int MAXM=10;

struct data{

int fi,se;

data( int fi\_=0 , int se\_=0 ){

fi=fi\_;se=se\_;

}

friend bool operator == ( const data &a , const data &b ){

return a.fi==b.fi && a.se==b.se;

}

};

int Plast,last,top;

int de[MAXM];

int tmp[MAXM];

data ar[MAXN];

data st[MAXN];

data postfix[MAXN];

string s;

void init(){

de[0]=1;

for( int i=1 ; i<MAXM ; i++ )

de[i]=de[i-1]\*10;

}

void read(){

for( int i=0 ; i<sz(s) ; i++,last++ )

if( s[i]=='(' )

ar[last]=data(1,1);

else if( s[i]==')' )

ar[last]=data(1,-1);

else if( s[i]=='+' )

ar[last]=data(2,1);

else if( s[i]=='-' )

ar[last]=data(2,2);

else if( s[i]=='\*' )

ar[last]=data(2,3);

else if( s[i]=='/' )

ar[last]=data(2,4);

else{

int j,k,sum=0;

for( k=0,j=i ; j<sz(s) && s[j]>='0' && s[j]<='9' ; j++,k++ )

tmp[k]=s[j]-'0';

k--;

for( int d=0 ; k>=0 ; k--,d++ )

sum+=de[d]\*tmp[k];

ar[last]=data(3,sum);

i=j-1;

}

//~ for( int i=0 ; i<last ; i++ )

//~ cout << ar[i].fi << ' ' << ar[i].se << endl;

}

void push( data cur ){

if( cur==data(1,1) )

st[++top]=cur;

else if( cur.se==1 || cur.se==2 ){

while( top && ( st[top]==data(2,3) || st[top]==data(2,4) ) )

postfix[++Plast]=st[top--];

st[++top]=cur;

}

else

st[++top]=cur;

}

void pop(){

while( !(st[top]==data(1,1)) )

postfix[++Plast]=st[top--];

top--;

}

void toPostfix(){

for( int i=0 ; i<last ; i++ )

if( ar[i].fi==3 )

postfix[++Plast]=ar[i];

else if( ar[i]==data(1,1) || ar[i].fi==2 )

push(ar[i]);

else

pop();

while(top)

postfix[++Plast]=st[top--];

}

void write( data cur ){

if( cur==data(1,1) )

printf("(");

else if( cur==data(1,-1) )

printf(")");

else if( cur==data(2,1) )

printf("+");

else if( cur==data(2,2) )

printf("-");

else if( cur==data(2,3) )

printf("\*");

else if( cur==data(2,4) )

printf("/");

else

printf("%d",cur.se);

}

int value( int a , int o , int b ){

if( o==1 )

return a+b;

if( o==2 )

return a-b;

if( o==3 )

return a\*b;

return a/b;

}

void process(){

for( int i=1 ; i<=Plast ; i++ )

if( postfix[i].fi==3 )

st[++top]=postfix[i];

else{

int tmp=value(st[top-1].se,postfix[i].se,st[top].se);

st[--top]=data(3,tmp);

}

cout << st[1].se << endl;

}

int main(){

cin >> s;

init();

read();

toPostfix();

for( int i=1 ; i<=Plast ; i++ )

write(postfix[i]);

cout << endl;

process();

return 0;

}