#include<bits/stdc++.h>

using namespace std;

int precendents(char op){

if(op=='+' || op=='-'){

return 1;

}

else if(op=='\*' || op=='/'){

return 2;

}

else{

return 2;

}

}

void solve(string s){

stack<string> post;

stack<string> pre;

stack<char> ops;

for(int i=0;i<s.size();i++){

char ch = s[i];

if(ch=='('){

ops.push(ch);

}

else if((ch>='0' && ch<='9') || (ch>='a' && ch<='z') || (ch>='A' && ch<='Z')){

string s1,s2;

post.push(s1+ch);

pre.push(s2+ch);

}

else if(ch==')'){

while(ops.top()!='('){

char op = ops.top();

ops.pop();

string post2 = post.top();

post.pop();

string post1 = post.top();

post.pop();

string res1 = post1+post2+op;

post.push(res1);

string pre2 = pre.top();

pre.pop();

string pre1 = pre.top();

pre.pop();

string res2 = op+pre1+pre2;

pre.push(res2);

}

ops.pop();

}

else if(ch=='+' || ch=='-' || ch=='\*' || ch=='/'){

while(ops.size()!=0 && ops.top()!='(' && precendents(ch)<=precendents(ops.top())){

char op = ops.top();

ops.pop();

string post2 = post.top();

post.pop();

string post1 = post.top();

post.pop();

string res1 = post1+post2+op;

post.push(res1);

string pre2 = pre.top();

pre.pop();

string pre1 = pre.top();

pre.pop();

string res2 = op+pre1+pre2;

pre.push(res2);

}

ops.push(ch);

}

}

while(ops.size()!=0){

char op = ops.top();

ops.pop();

string post2 = post.top();

post.pop();

string post1 = post.top();

post.pop();

string res1 = post1+post2+op;

post.push(res1);

string pre2 = pre.top();

pre.pop();

string pre1 = pre.top();

pre.pop();

string res2 = op+pre1+pre2;

pre.push(res2);

}

cout<<post.top()<<endl;

cout<<pre.top()<<endl;

}

int main(){

string s;

getline(cin, s);

solve(s);

}