**Expression conversion and evaluation.**

#include <iostream>

using namespace std;

class convert{

char expr[20]="",out[20]="",stacks[20];

int top=-1,top1=-1,j=0,stacks1[20];

public:

void accept(){

cout<<endl<<"Enter your infix expression:";

cin>>expr;

}

void push(char ch){

stacks[++top]=ch;

}

char pop(){

return stacks[top--];

}

int precedence(char ch){

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

return 1;

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

return 2;

}else{

return 0;

}

}

void conversion(){

int i=0;

while(expr[i]!='\0'){

if(expr[i]=='(' || expr[i]=='{' || expr[i]=='['){

push(expr[i]);

}else if(expr[i]==')' || expr[i]=='}' || expr[i]==']'){

if(expr[i]==')'){

while(stacks[top]!='('){

out[j++]=pop();

}

}else if(expr[i]=='}'){

while(stacks[top]!='{'){

out[j++]=pop();

}

}else if(expr[i]==']'){

while(stacks[top]!='['){

out[j++]=pop();

}

}

char temp=pop();

}else if(expr[i]=='+' || expr[i]=='-' || expr[i]=='\*' || expr[i]=='/'){

if(precedence(stacks[top])>=precedence(expr[i])){

out[j++]=pop();

push(expr[i]);

}else{

push(expr[i]);

}

}else{

out[j++]=expr[i];

}

i++;

}

if(top!=-1){

while(top>=-1){

out[j++]=pop();

}

}

out[j++]='\0';

cout<<endl<<"Postfix expression is:";

for(int i=0;out[i]!='\0';i++){

cout<<out[i];

}

}

int popInt(){

return stacks1[top1--];

}

void pushInt(int num){

stacks1[++top1]=num;

}

void evaluate(){

int op1,op2;

int i=0,num;

while(out[i]!='\0'){

if(out[i]=='+'){

op2=popInt();

op1=popInt();

pushInt(op1+op2);

}else if(out[i]=='-'){

op2=popInt();

op1=popInt();

pushInt(op2-op1);

}else if(out[i]=='\*'){

op2=popInt();

op1=popInt();

pushInt(op1\*op2);

}else if(out[i]=='/'){

op2=popInt();

op1=popInt();

pushInt(op2/op1);

}else{

cout<<endl<<"Enter integer value for:"<<out[i];

cin>>num;

pushInt(num);

}

i++;

}

cout<<endl<<popInt();

}

};

int main()

{

convert c1;

c1.accept();

c1.conversion();

c1.evaluate();

return 0;

}