데이터 구조 과제

학번: 12170584

이름: 이진호

**Infix to Postfix**

**문제정의:**

1. Infix 형태로 표현된 수식을 stdin에서 입력받아 postfix 형태로 바꾼 후 stdout으로 출력한다.
2. stdin에서 하나의 수식 입력받고 postfix가 stdout으로 출력된 후에도 계속해서 다른 수식을 입력받을 수 있도록 하며, 입력 수식이 없는 경우 종료하도록 해야 한다.

**코드 설명:**

1. **우선 순위 연산자 계산**

int \_compare(char item){

If(item == ‘+’ || item==’-‘) return 1; //연산자가 +,-이면 우선순위 1

else if(item == ‘\*’ || item==’/’) return 2; //연산자가 \*,/이면 우선순위 2

else return -1; //그이외의 경우에는 우선순위 -1

}

1. **Infix를 Postfix로 출력**

//infix를 postfix로 바꾸는 함수

void infixToPostfix(string infix){

string result;

//infix의 길이 만큼 동작

for(int i=0;i<infix.length();i++){

//infix의 처음에 -가 있을때,

if(infix[0] == '-' && i==0){

result+='-';

continue;

}

//infix의 처음에 +가 있을때,

else if(infix[0] == '+' && i==0){

continue;

}

//infix에 숫자가 있을때,

else if(infix[i] >= '0' && infix[i] <='9'){

result +=infix[i];

}

//infix의 i번째에 (가 있을때,

else if(infix[i]=='('){

\_push(infix[i]);

//양수 음수 표현

if(infix[i+1] == '-'){

i++;

result+=infix[i];

}

else if(infix[i+1] == '+'){

i++;

}

}

//infix의 i번째에 )가 있을때,

else if (infix[i] == ')'){

//스택에 ( 를 찾을때 까지 pop하여 result에 추가

while(stack[top] != '(' && top !=-1){

char oper = stack[top];

\_pop();

result =result + " "+ oper;

}

// ( 연산자를 pop

if(stack[top]=='('){

\_pop();

}

}

//infix에 연산자가 있을때, 연산자 비교

else if(infix[i] == '\*' || infix[i] == '/' || infix[i] == '+'

|| infix[i] == '-'){

//스택에 있는 연산자와 infix의 i번째 연산자의 우선순위 비교

while(top!=-1 && \_compare(infix[i]) <= \_compare(stack[top])){

//스택에 있는 연산자 pop

result=result+" "+\_pop();

}

result+=" ";

//infix 연산자 push

\_push(infix[i]);

//양수 음수 표현

if(infix[i+1] == '-'){

i++;

result+=infix[i];

}

else if(infix[i+1] == '+'){

i++;

}

}

//에외 처리

else{

cout << infix[i]<<" 는 적절하지 않은 피연산자입니다.\n";

exit(1);

}

}

//스택에 들어있던 연산자 모두 pop하여 출력

while(top!=-1){

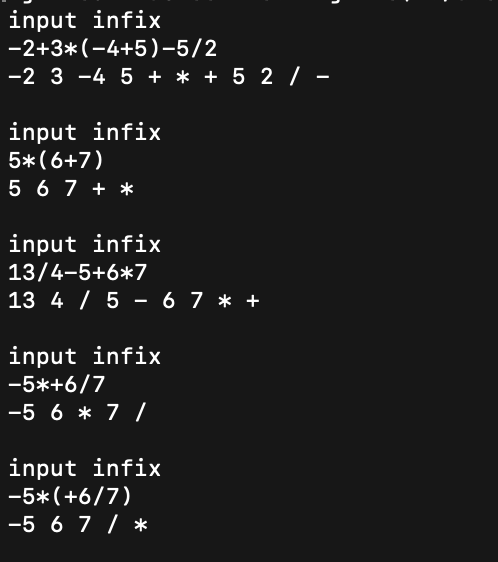
result=result+" "+\_pop();

}

cout << result << endl;

}

**출력 결과:**



**결론- 스택을 이용하여 infix를 postfix로 바꾸는 작업을 할 수 있었다.**