|  |  |
| --- | --- |
| sym01_s | **윈도우프로그래밍**  **LAB 03** |
| **분반** | **001** |
| **학번** | **2010044** |
| **이름** | **박진희** |

**# 실습 문제 1**

#1 balanced.cpp 프로그램 소스(수정한 결과 소스)

#include <iostream>

#include <strstream>

using namespace std;

struct Stack {

char\* store;

int capacity;

int top;

};

int create(Stack& s, int size = 100) {

s.capacity = size;

s.top = 0;

return (s.store = new char[size]) != 0;

}

void push(Stack& s, char val) {

if (s.top < s.capacity)

s.store[s.top++] = val;

else

cerr << "Error: stack full, can't push" << endl;

}

char pop(Stack& s) {

if (s.top > 0)

return s.store[--s.top];

else {

cerr << "Error: pop from an empty stack" << endl;

return 0;

}

}

int isempty(Stack& s) {

return s.top == 0;

}

int isopeningparen(char c) {

if (c == '(' || c == '{' || c == '[')

return 1;

else

return 0;

}

int isclosingparen(char c) {

if (c == ')' || c == '}' || c == ']')

return 1;

else

return 0;

}

int matching(char c1,char c2) {

switch(c2){

case ')':

if (c1 == '(')

return 1;

case ']':

if (c1 == '[')

return 1;

case '}':

if (c1 == '{')

return 1;

default:

return 0;

}

}

int check\_balance(char expr[]) {

Stack s;

create(s);

istrstream is(expr);

int i = 0;

char c;

while (expr[i]) {

is.get(c);

if (isopeningparen(c))

push(s, c);

else if (isclosingparen(c) && !matching(pop(s), c)) {

cout << ":MISMATCHING!!";

return 0;

}

i++;

}

if (isempty(s))

cout << ":BALANCED!!";

else

cout << ":MISMATCHING!!";

}

#define LINESIZE 1000

int main() {

char line[LINESIZE];

while (cin.get(line, LINESIZE)) {

cout << line ;

char c;

cin.get(c);

check\_balance(line);

cout << endl;

}

}

#2 실행 화면

