**Student Number: 202135760**

**Student Name: 김현지**

**Problem 5**

* **Screenshots**

**텍스트, 스크린샷, 폰트, 디스플레이이(가) 표시된 사진

자동 생성된 설명**

* **Code**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<ctype.h>

#define X\_MAXLEN 10001 //최대 10000자

#define Z\_MAXLEN 101 //최대 100자

int dynamic\_programming(char[], char[]);

void main() {

int num;

char x[X\_MAXLEN];

char z[Z\_MAXLEN];

scanf("%d", &num);

int\* output = (int\*)malloc(sizeof(int\*) \* num); //결과값

memset(output, 0, sizeof(int) \* num); //결과값 초기화

int xIsLower = true; //x가 lowercase로 이루어져 있으면 true, 아니면 false

int zIsLower = true; //z가 lowercase로 이루어져 있으면 true, 아니면 false

for (int i = 0; i < num; i++) {

do {

printf("[#%d] Enter lowercase alpahbetic characters: \n", i);

scanf("%s", &x);

scanf("%s", &z);

xIsLower = true;

zIsLower = true;

for (int stringIndex = 0; stringIndex < strlen(x); stringIndex++) {

if (isupper(x[stringIndex]))//x에 uppercase가 있으면 xIsLower에 false할당

xIsLower = false;

}

for (int stringIndex = 0; stringIndex < strlen(z); stringIndex++) {

if (isupper(z[stringIndex]))//z에 uppercase가 있으면 zIsLower에 false할당

zIsLower = false;

}

} while (!(xIsLower && zIsLower));//z, x에 upppercase가 있다면 다시 입력받기

output[i] = dynamic\_programming(x, z);

}

//결과

for (int i = 0; i < num; i++) {

printf("%d\n", output[i]);

}

}

int dynamic\_programming(char x[], char z[]) {

int width = strlen(x) + 1; //x 글자수 만큼 table 열 만들기 + 첫번째에 빈 열 추가

int height = strlen(z) + 1; //z 글자수 만큼 table 행 만들기 + 첫번째에 빈 행 추가

int\*\* table = (int\*\*)malloc(sizeof(int\*) \* height);//table 행 만들기

for (int index = 0; index < strlen(z) + 1; index++) {

table[index] = (int\*)malloc(sizeof(int) \* width);//table 열 만들기

}

int i = 0;

int j = 0;

//첫번째 행 1로 초기화, 첫번째 열 0으로 초기화

for (j = 0; j < width; j++) {

table[0][j] = 1;

}

for (i = 1; i < height; i++)

table[i][0] = 0;

for (i = 1; i < height; i++) {

for (j = 1; j < width; j++) {

if (x[j - 1] == z[i - 1]) {//같으면 표 [왼쪽+왼쪽 대각선 위] 값 가져오기

table[i][j] = table[i][j - 1] + table[i - 1][j - 1];

}

else//틀리면 표 왼쪽에서 값 가져오기

table[i][j] = table[i][j - 1];

}

}

int result = table[height - 1][width - 1];

//동적할당 해제

for (int index = 0; index < height; index++) {

free(table[index]);

}

free(table);

return result;

}

**Problem 2**

* **Screenshots**
* **Code**