

객체지향프로그래밍설계

1차 과제 레포트

학 과: 컴퓨터정보공학부

담당교수: 공영호

학 번: 2018202030

성 명: 박지용

Assignment 1-1

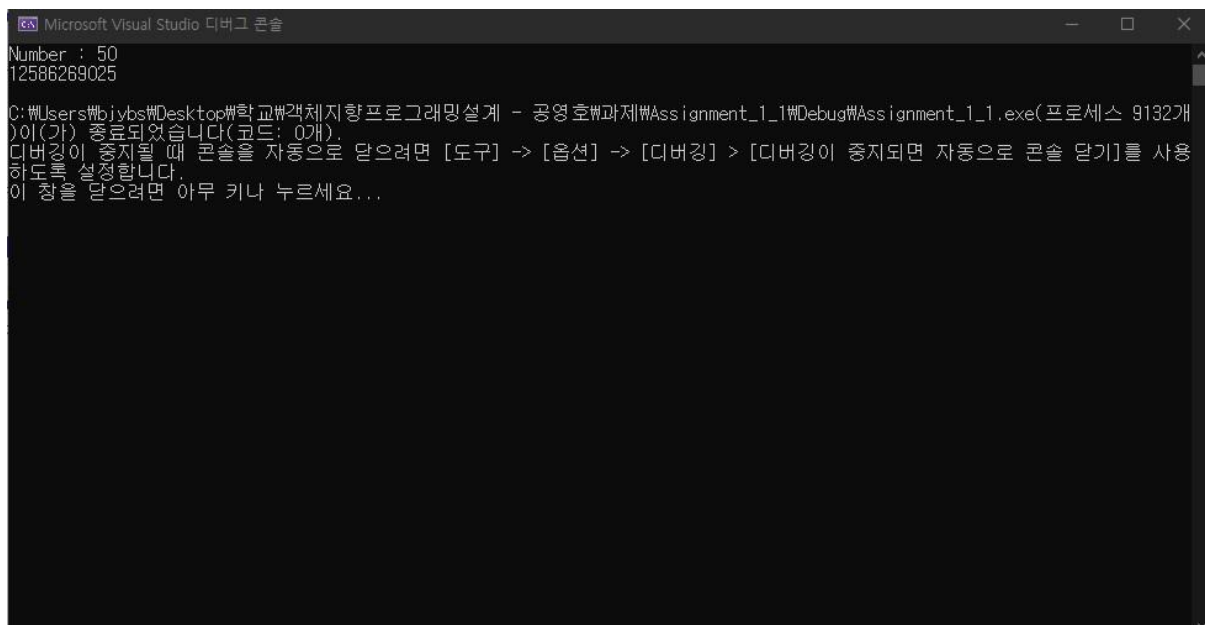
$0 < N < 51$ 의 N 에 대해서 피보나치 수열을 만들어라. 피보나치 수열은 다음과 같은 조건을 만족시킨다.

$$F_1 = F_2 = 1$$

$$F_n = F_{n-1} + F_{n-2}$$

첫째항과 둘째항은 1이고 3째 항부터 $F_n = F_{n-1} + F_{n-2}$ 조건이 적용된다. 또한, n 이 50일 경우 표현해야 하는 숫자의 범위가 늘어나기 때문에 그 점을 고려해서 long long 변수를 이용하여 배열을 이용하여 각 N 번째 항을 크기 51인 long long 배열에 저장한다.

결과화면



```
Microsoft Visual Studio 디버그 콘솔
Number : 50
12586269025
C:\Users\biybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\Assignment_1_1\Debug\Assignment_1_1.exe(프로세스 9132개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

고찰

처음에 문제를 풀었을 때 재귀함수를 이용하여 풀었는데 40부터 결과값이 안 나오는 것을 발견했다. 처음에는 코드가 문제가 없어서 어떤 문제인지 몰랐는데 간단하게 재귀함수 호출이 되는 순서를 정리하면서 재귀함수 사용시 함수 호출이 기하 급수적으로 늘어나는 것을 알게 되었고 간단하게 배열에 1항부터 50항까지 계산해서 대입하는 방식으로 문제를 풀었다. 다시 풀고 50을 입력해보니 또 음수값 나오는 것을 발견했고 int형 배열이 표현 가능한 수를 넘은 것을 바로 알아차려서 long long으로

배열의 자료형을 바꿔주고 정상적인 결과값을 얻을 수 있었다. 간단한 문제지만 자료형과 재귀함수를 사용했을 경우를 생각해주어야 했다.

Assignment 1-2

2차원 배열의 size를 입력 받고 방향을 입력 받는다. 입력 받은 방향으로 더해가면서 방향에 있는 원소들을 제외하고 다른 원소들은 0이된다. 먼저, matrix size N을 입력 받는다. 입력을 받고 모든 원소의 값이 1인 matrix를 출력한다. M을 입력 받고 방향에 따라 더하기가 된 matrix를 출력한다.

Up, right, left, down의 함수는 각 행, 각열의 모든 원소들의 합을 sum에 저장 후 각 대응되는 열 혹은 행의 원소들에 sum을 저장하고 그 외의 모든 원소들은 0으로 초기화 하는 방법으로 각 함수를 정의했다.

예를 들어 Up 의 함수의 경우 for문을 마지막 행 (size - 1)행부터 0행까지 각 행의 열들의 합을 sum에게 저장하고 for문의 마지막 행인 0행에 sum을 저장한다. 그리고 마지막으로 UP방향인 0행의 원소들을 제외한 1행부터 size - 1행까지 모든 원소를 0으로 초기화를 시킨다.

1	1	1
1	1	1
1	1	1

<size가 3인 matrix>

각 행의 원소들을 sum에 저장하고 up방향인 0행의 0열부터 size - 1열인 2열까지 각 저장한다.

결국 sum = 3

3	3	3
1	1	1
1	1	1

3	3	3
0	0	0
0	0	0

<최종 matrix>

마지막으로 up방향인 0행을 제외한 모든 원소를 0으로 초기화 한다. 위 matrix가 최종 matrix이다.

위 4개의 함수와는 다르게 downLeft, upRight함수는 비슷하지만 다르게 작동한다. 먼저 downleft를 예로 들면 downleft방향의 원소들은 각각 자신의 대각선 방향의 원소들은 더한 값을 취한다. 그래서 3행3열 matrix를 예로 들면

1	1	1
1	1	1
1	1	1

변수 x, y를 선언하고 x, y는 각각 노란색의 원소들을 지나가게 증감연산자를 이용하여 loop안에서 황색 위치를 가르키게 만든다. 초기 x, y의 값은 0이고 y가 2가 되었을때와 그 전을 나눠서 조건을 만들었다. y가 2가 되기전 즉, 0과 1일 경우에는 증감식은 y++이다. y가 2일 경우 x++이다.

그 다음 x, y가 가르키는 지점에서 대각선 방향으로 downleft 방향으로 원소들의 합을 sum에 저장한다. x, y가 [0][2]일 경우에 대각선 방향으로 이동하게 되는 증감식은 [x--][y++]이고 [0][2]의 값과 [1][1], [2][0]의 값을 sum에 저장하고 마지막 x, y지점 [x][y]지점([2][0])에 sum을 저장한다.

3	1	1
3	1	1
3	3	3

마지막으로 downleft방향 황색의 부분은 sum의 값인 3이 되고 황색 부분을 제외한 다른 원소는 0으로 초기화를 해준다.

3	0	0
3	0	0
3	3	3

최종 matrix의 모습은 위와 같고 upright함수는 downleft의 함수의 정 반대로 실행되는 함수이다.

결과화면

```

Microsoft Visual Studio 디버그 콘솔
Array Size(0<N<10) : 5
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

Shift All Direction : 5
1 2 3 4 5
0 0 0 0 4
0 0 0 0 3
0 0 0 0 2
0 0 0 0 1

C:\Users\Wbjybs\Desktop\공학 교재\체제향 프로그래밍 설계 - 공영 호\과제\Assignment_1_2\Debug\Assignment_1_2.exe( 프로세스 24028
개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...

```

고찰

Up, right, left, right함수는 비교적 쉽게 구현이 가능했지만 대각선으로 "값을 밀어내는" 함수는 처음에 구현하기가 쉽지 않았다. For문 중첩을 이용하여 처음에 비슷한 방식으로 풀려고 했으나 기준이 되는 원소가 같은 행으로 자다가 "꺼어져" 아래로 내려가기 때문에 조건문을 걸어주지 않는 이상 중첩for문으로 바로 해결하기가 쉽지 않았다. 그래서 열을 증가 시켜서 기준원소가 같은 행에 있는 경우와 같은 열의 있는 경우를 조건문으로 나누어서 구현을 하였다.

Assignment 1-3

N 크기의 matrix를 입력 받고 그 matrix를 sort(정렬)하는 문제이다. Matirx에 모든 원소는 0보다 크고 100보다 작은 random number로 이루어져있다. 원소들은 실행될 때 마다 다른 원소들이여야 하며 직접 sorting 함수를 구현하고 ascending order 오름차순으로 정렬이 되어야 한다. 먼저, array size를 입력 받고 2차원 배열 동적할당을 이용해 size 크기의 2차원 배열을 만들어 낸다. 그리고 random 한 값이 들어가 있는 2차원 배열을 출력하고 sorting을 거친 후 정렬된 배열을 출력한다.

매번 다른 random한 값을 위해서 srand함수를 이용했다. srand함수는 매개변수로 unsigned int형의 seed값을 받고 rand 함수의 사용될 수를 초기화 하는 함수이다. 매번 다른 random number가 나오게 하려면 seed값을 시간을 전달해주면 된다. `srand((unsigned int)time(NULL));`

sorting중에서 나는 bubble sort를 이용하여 정렬을 하였다. Bubble sort는 인접해 있는 원소들끼리 비교하여 원소의 끝까지 비교하는 방식이다. 그렇게 인접한 원소 중에 큰 것을 오른쪽으로 보내면 우측원소는 정렬이 된 상태가 되고 중첩for문을 이용해 정렬이 된 원소를 제외하고 다시 처음부터 인접한 원소들끼리 비교를 한다.

2차원 배열에서 bubble sort를 다음과 같이 구현했다. 처음에 제일 문제였던 것은 2차원 배열에서 다음행으로 넘어갈 때 두 원소를 비교하는 것이 어려웠다. 내가 생각해 낸 방법은 2차원 배열을 1차원 배열처럼 생각 을해서 푸는 방법이었다. 예를 들면 3 X 3배열은 [0][0]부터 [2][2]까지 원소가 있고 2차원 배열을 1차원 적으로 봤을 때 0번째 원소부터 8번째 원소까지 있다고 생각을 하였다. 그렇게 변수 j를 0으로 초기화 한 후 증감연산자를 통하여 1씩 증가시키면서 행의 index값을 j / N 으로 열의 index를 j / N 으로 하고 bubble sort를 구현하였다. 3 X 3배열에서 [2][1]은 `array[j / size][j % size]` 즉 `array[7 / 3][7 % 3]`, `array[2][1]` 이런 식으로 j를 이용하여 인접한 원소처럼 표현하여 구현하였다.

결과화면

```
Microsoft Visual Studio 디버그 콘솔
Array Size(0<N<11) : 5
58 37 77 83 21
75 26 35 88 41
100 76 94 44 69
2 56 92 8 99
76 56 75 57 62

2 8 21 26 35
37 41 44 56 56
57 58 62 69 75
75 76 76 77 83
88 92 94 99 100

C:\Users\biybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\Assignment_1_3\Debug\Assignment_1_3.exe(프로세스 12776)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

```
Microsoft Visual Studio 디버그 콘솔
Array Size(0<N<11) : 5
34 75 37 51 12
36 76 43 39 8
38 89 100 62 100
52 78 88 46 10
39 69 36 52 83

3 10 12 34 36
37 39 43 46 51
52 62 62 89 75
76 78 83 86 88
39 89 98 100 100

C:\Users\biybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\Assignment_1_3\Debug\Assignment_1_3.exe(프로세스 4992)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

결과화면으로 똑 같은 size 배열이지만 random값이 다른 것을 볼 수 있다.

고찰

처음 2차원 배열을 정렬하는데에 있어서 생각을 좀 해야했다. 중첩for문 만을 이용해 원소들을 비교해서 sorting을 구형했지만 행의 끝과 다음 행의 시작 원소가 비교가 안되고 결론적으로 각 행을 sorting이 되는 정렬알고리즘을 구현해서 처음에는 잘 되는 줄 알았지만 모든 원소에 대해서 정렬이 안되는 것을 악세 되었고 1차원적으로 생각을 해서 bubble sort를 구현하게 되었다. 동적할당을 사용하기 visual studio가 아닌 xcode로 컴파일을 했을때는 VLA(variable length array)지원이 되어서 VLA

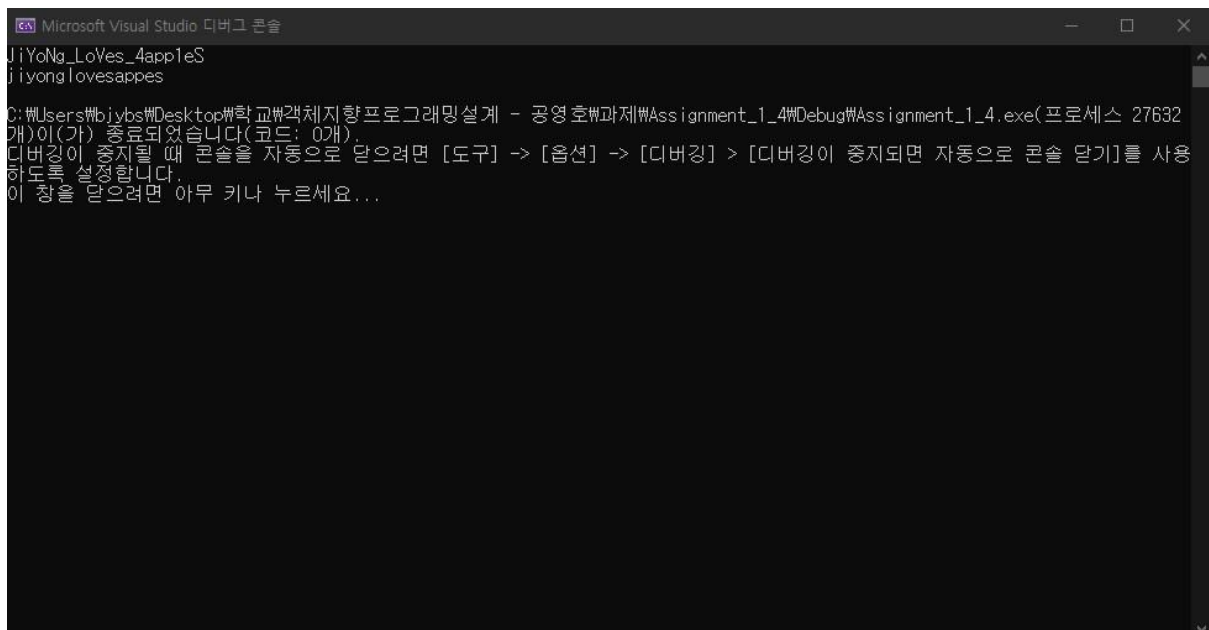
를 이용해서 배열을 선언하려고 하였지만 VLA가 사용이 안되어서 동적할당을 이용하여 2차원 배열을 선언하였다.

Assignment 1-4

문자열을 입력 받고 문자열에서 알파벳만 출력하되 lower case로 출력하는 문제이다. 알파벳 이외의 문자는 출력을 안 한다.

먼저 문자 cin.get() 함수를 이용하여 문자 c에 입력 받은 문자를 하나씩 i = 0번째 항부터 엔터가 나오기 전까지 배열의 i번째 항에 대입한다. 루프가 끝나면 마지막 i자리에 널문자를 넣어주어 문자열의 끝을 알려준다. 알파벳을 확인 하기 위해서 isalpha()함수를 사용하였고 만약 isalpha()함수가 참이면 tolower함수를 이용하여 lower case로 변환해 준 후 (char)형으로 변환 후 출력한다.

결과화면



```
Microsoft Visual Studio 디버그 콘솔
jiYong_LoVes_4appleS
jiyong loves apples

C:\Users\bjybs\Desktop\학 교\객체지향 프로그래밍 설계 - 공영호\과제\Assignment_1_4\Debug\Assignment_1_4.exe(프로세스 27632
개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

고찰

문자열을 입력 받을 때 cin을 사용하지 않은 이유는 whitespace를 포함해서 온전한 문자열을 저장하고 싶었기 때문이다. 그래서 생각한 방법이 while(cin >> str)을 사용하는 방법이었는데 이 방법은 동일하게 white space를 cin함수가 받을 수 없기 때문에 문자열이 받아질 수 가 없었다. 그래서

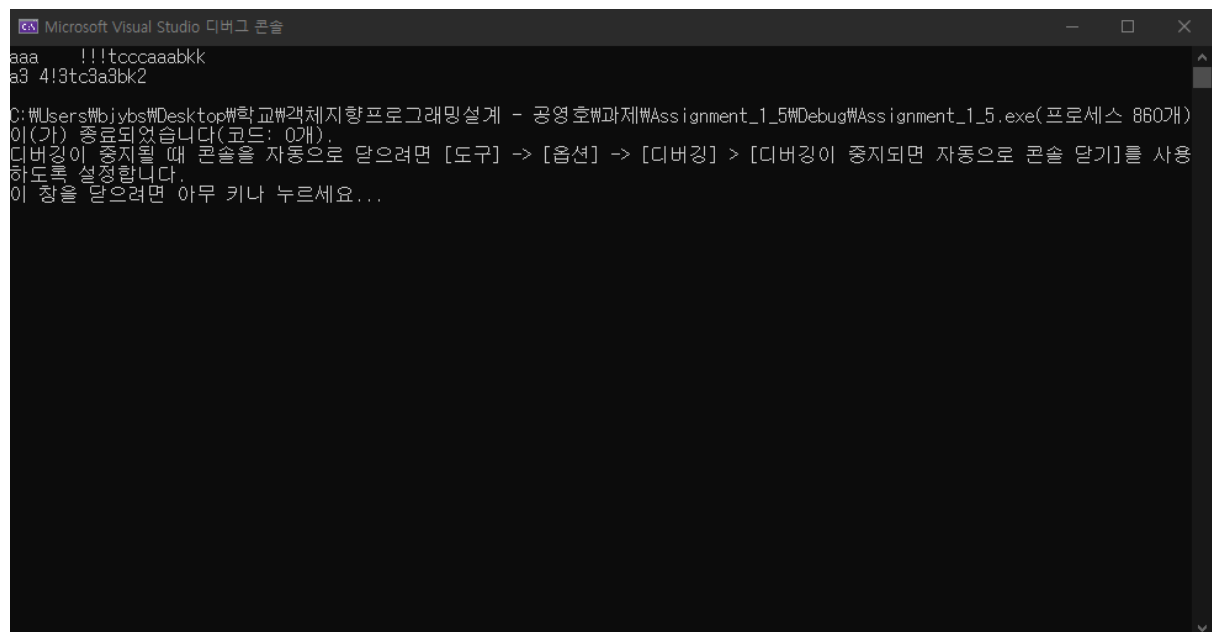
cin.get()함수를 사용하였다. 처음 짰 코드는 문자열에 저장조차 하지 않고 cin.get()함수로 문자를 받고 바로 알파벳인지 확인하고 lower case로 출력하는 방법을 사용하였으나 문제의 조건에 문자열을 받으라고 나와있어서 수정을 하였다.

Assignment 1-5

반복되는 문자를 축소하는 문제이다. 반복되는 문자를 문자와 반복되는 횟수로 표현하는 방식이다. 먼저 문자열을 받아서 축소된 문자열에 저장하여 출력하는 방식이다. 2번 이상 반복되는 문자에 대해서 축소한다.

먼저 cin.get()함수를 이용한 문자열을 받고 문자열의 끝을 표기해준다. 그 다음 checkc라는 char형 변수에 문자열에서 처음으로 나오는 문자를 저장해 준 후 "기준문자"로 활용한다. 만약 다음 index에 있는 문자가 기준문자와 동일하다면 count를 +1해주고 아니라면 converted_str문자열에 차례대로 문자와 반복되는 숫자를 넣어준다. 여기서 주의해야 할 점은 1번 반복되는 문자는 예를 들어 a1이 아니라 그냥 a로 저장이 되어야 한다.

결과화면



```
Microsoft Visual Studio 디버그 콘솔
aaa !!!tcccaaabkk
a3 4!3tc3a3bk2
C:\Users\Wbjybs\Desktop\공학교과객체지향프로그래밍설계 - 공영호\과제\Assignment_1_5\Debug\Assignment_1_5.exe (프로세스 860개)
이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

고찰

문제를 제대로 안 읽고 처음에 2번 이상 반복되는 숫자가 아닌 1번만 나와도 a1이런식으로 문자열에 저장되게 하는 방식으로 프로그램을 구현했다. 문제를 다시 읽고 고친 점 2가지는 1번 반복되는 문자는 숫자가 없고 변형된 새로운 문자열에 저장을 하여서 출력이 되도록 하였다. 원래는 문자열에 따로 저장하지 않고 바로바로 출력이 되는 방식으로 구현을 하였지만 문제에 output the converted string이라고 나와있어서 코드를 수정하였다.

Assignment 1-6

심플 체크섬을 구현하는 문제이다. 문제는 크게 3부분으로 나누어져 있다. Sender, Transmission_Process 그리고 Receiver로 구성 되어있다. Sender는 4개의 숫자를 입력을 받고 체크섬을 계산한 뒤 5개의 숫자를 int형 배열에 저장한다. 그리고 출력을 하고 Transmission_Process함수에 배열을 전달한다. Transmission_Process 함수에서는 매개변수로 전달받은 const char*배열에 40퍼센트의 확률로 Error를 생성하는 함수이다. Const char*형식은 데이터를 바꿀 수 없기 때문에 새로운 char형 배열에 전달받은 배열을 복사한 후 error를 생성한다. Receiver는 Transmission_Process에 의해서 error가 생성된 배열을 매개변수로 받아 체크섬이 정확한지를 판단하는 함수이다.

Sender 함수에서 cin.get()을 이용하여 문자를 입력 받는다 while이 끝나고 바로 널문자를 넣는 것이 아니라 입력 받은 4개의 숫자를 더하여 10으로 나눈 나머지를 배열에 저장해 준 후 널문자를 삽입해 준다. Transmission_Process함수에서 manipulatedData문자열에 매개변수로 받은 Data문자열을 복사하고 srand((unsigned int)time(NULL))을 이용하여서 난수를 initialize한다. 먼저 체크섬을 포함한 0부터 4번째 원소들 중 어디를 manipulate할지 정해야 하기 때문에 rand() % 5를 이용하여 manipulatedData[rand() % 5]에 error를 생성한다. 40퍼센트의 확률을 임의대로 0부터 9까지의 숫자 중 0부터 3까지 4개의 숫자가 나왔을 때 error가 생성되게 한다. 또한 manipulatedData[rand() % 5]에 있는 숫자가 error를 만들어 내기 위한 숫자와 동일하다면 같은 숫자가 아닐 때까지 random number를 받는다. Receiver함수에서 매개변수를 포인팅 하는 const char *p변수를 만들고 data의 널문자에 다르게 ++p를 해준다. 감소시켜가면서 4번째 항을 checksum int형 변수에 저장하고 0부터 3항 원소를 더한 checksum가 비교하고 프로그램이 종료된다.

결과화면

```
Microsoft Visual Studio 디버그 콘솔
Data : 4741
Send data : 47416
Received datd : 47316
6!=5 Error!

C:\Users\bjybs\Desktop\학교\백제지향프로그래밍설계 - 공영호\과제\Assignment_1_6\Debug\Assignment_1_6.exe(프로세스 28564)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

```
Microsoft Visual Studio 디버그 콘솔
Data : 4741
Send data : 47416
Received datd : 47416
6==6 Error is not 0curred!

C:\Users\bjybs\Desktop\학교\백제지향프로그래밍설계 - 공영호\과제\Assignment_1_6\Debug\Assignment_1_6.exe(프로세스 1016개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

고찰

40퍼센트의 error확률을 구현해 내는 것이 어려웠다. 10개의 숫자 중 어떻게 40프로의 확률을 생성해 낼까 생각이 잘 안 났지만 임의로 0부터 3까지의 숫자가 나오는 확률을 생각하고 정해줬다. 또한 원래 숫자와 같을 경우를 생각해서 같은 숫자가 안 나올때까지 loop문을 못 빠져나오게 하였다. 마지막 Receiver에서 어떻게 const char*형을 이용하여서 checksum이 맞는지를 확인할까 고민을 많이 했다. 포인터를 이용해 간접적으로 매개변수로 받은 문자열을 참조하여 checksum을 계산하여 비교를 하였다. 함수도 많이 생성해야하고 40프로의 확률이라는 부담감이 컸지만 논리적이게 생각하고 구현

하였다.

Assignment 1-7

다항식이 valid한지 판별하는 문제이다. 다항식의 괄호, +, -, /, *, 숫자와 x를 제외하고 모든 인
풋은 제외된다. 모든 괄호는 짝을 지어야 한다. 곱셈은 skip하면 안 된다 $3x(x)^3 * x(o)$.

다항식이 유효하면 integrity 유효하지 않으면 Faulty를 출력하고 프로그램은 종료된다. 나는
isNum, isOperator, isInput, Faulty함수 4개를 이용하여 구현을 하였다. 전체적인 틀은 인풋을 문자열
저장하고 하나씩 검사를 하는 방식이다. 검사하는 문자에 대해서 그 전 원소와 그 다음 원소를 검사하
는 방식이다. isNum함수는 문자가 숫자인지 판별하는 함수이다. isOperator은 연산자인지 확인하고
isInput은 문자가 다항식의 괄호, +, -, /, *, 숫자와 x인지 확인하는 함수이다. Faulty는 마지막 faulty를
print해주는 함수이다.

검사하는 문자의 그 전 항과 그 다음항에 올 수 있는 문자와 올 수 없는 문자들을 표로 만들어
서 모든 조건을 조건문으로 예외처리를 하였다. 괄호의 짝을 맞추는 일은 parenthesisIn이라는 int형
변수를 이용하여서 '('괄호가 들어오면 +1 ')'괄호가 들어오면 -1을 하는 방법으로 구현하였다.

뒤에 오는 것이 가능한 표

	()	Operator	Number	x
(O	O	X	O	O
)	X	O	O	X	X
Operator	O	X	X	O	X
Number	X	O	O	O	X
x	X	O	O	X	X

앞에 오는 것이 가능한 표

	()	Operator	Number	x
(O	X	X	X	X
)	O	O	X	O	O
Operator	O	O	O	O	O
Number	O	X	O	O	X
x	O	X	O	X	X

이 두 표를 토대로 matrix[i]가 검사하는 문자라면 matrix[i - 1]과 matrix[i + 1]을 검사해 만약에 앞 혹은 뒤에 오면 안되는 문자가 있다면 뒤에 있는 문자들을 검사 할 필요도 없이 "faulty"를 출력한다.

예를 들면 operator 뒤에는 '('와 operator가 올 수 없고 앞에는 '('와 ')'가 올 수 없다. 따라서 조건문을 만들면 아래와 같다.

```
if(matrix[i - 1] == '(' || isOperator(matrix[i - 1]))

    Faulty

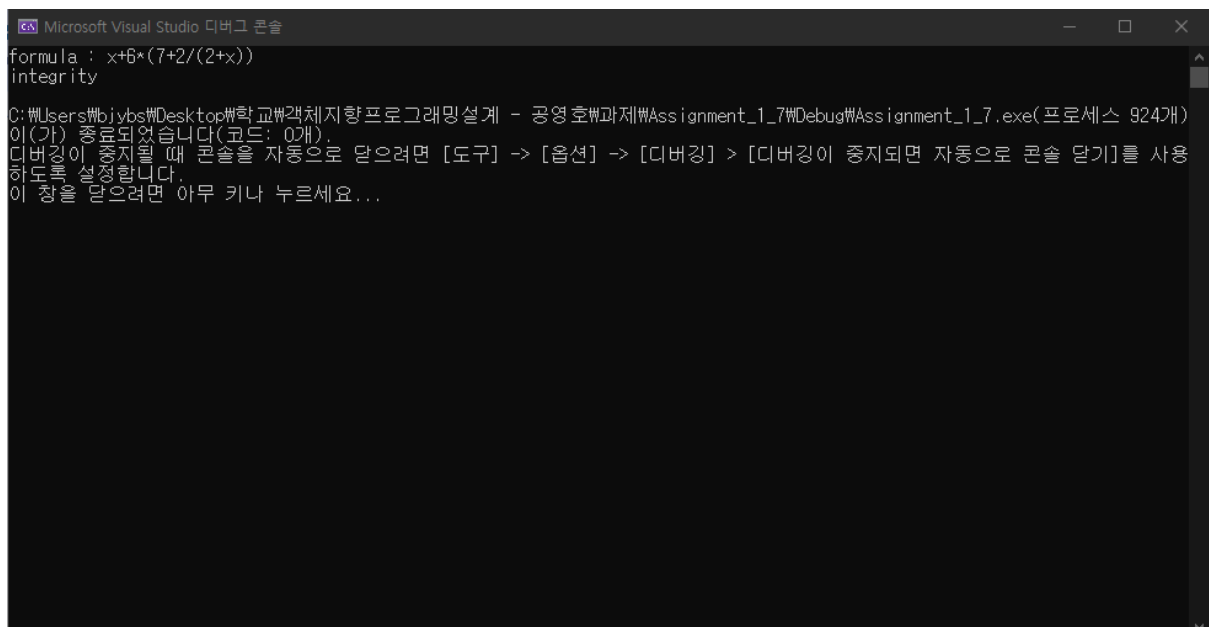
else if(matrix[i - 1] == '(' || (matrix[i - 1] == ')')

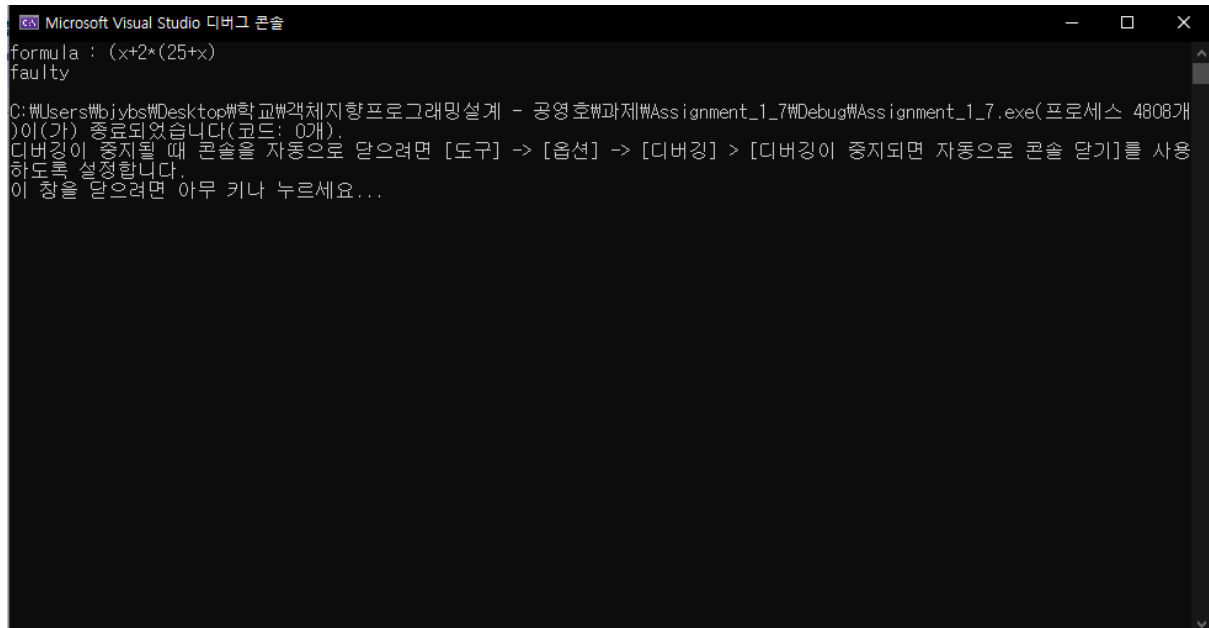
    Faulty

Else

    Integrity
```

결과화면





고찰

다른 구현방법이 많겠지만 나는 조건문을 이용해서 구현을 했다. 원래는 스택구조를 이용하여 구현하려고 생각을 했다. 하지만 스택의 푸시와 팝을 구현하더라고 어떻게 구현해야 할지 모르겠어서 논리적으로 앞뒤에 뭐가 오면 안 되는지를 생각했다. 그렇게 처음에 생각을 해도 어떻게 구현할지 감이 안 와서 먼저 표를 그리면서 앞과 뒤에 뭐가 오면 안 되는지를 파악하고 그것을 토대로 조건문을 이용해서 구현했다. 이 방법이 효율적인 방법으로는 생각이 되지는 않으나 내가 구현할 수 있는 가장 최선의 방법이라고 생각해서 이방법을 이용하였다.

Assignment 1-8

Matrix를 방향의 따라 a로 채워가는 프로그램이다. Matrix는 모두 '0'으로 초기화 되어있고 프로그램은 input을 받아 그 방향으로 a를 채워간다. 만약에 입력 받은 방향이 이미 'a'로 채워져 있으면 command는 취소가 된다. 또한 배열의 밖을 도달했을 때 똑같다. 만약 command가 행해졌을 때 console을 clean한다.

먼저 matrix를 생성하고 모든 원소를 0으로 초기화한다. Current_x, current_y변수를 이용해서 매번 커맨드를 실행한 후의 위치를 저장해준다. Left, right, up, down함수의 매개변수로 matrix와 current_x, current_y를 pass by reference로 넘겨준다. 그러면 각 함수에서 그 방향의 위치가 배열을 벗어나는지 혹은 이미 'a'가 채워져 있는지를 판단하고 그게 아니라면 그 방향으로 'a'를 채워주게 된다.

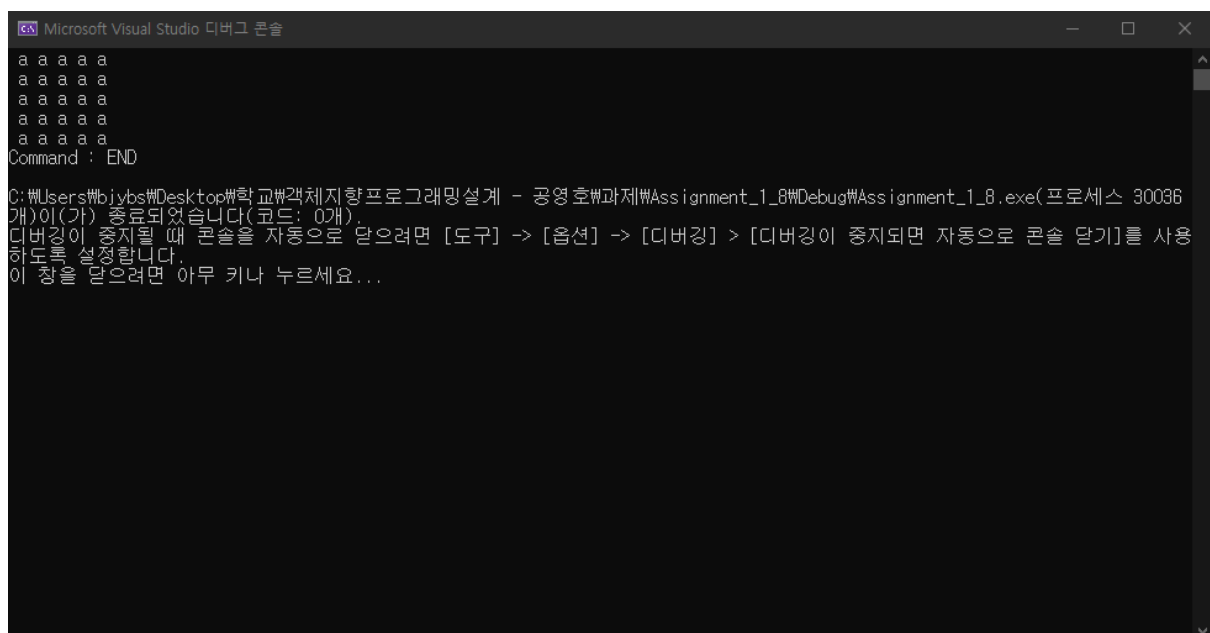
그리고 print함수를 통해 matrix를 출력하고 다시 command를 입력 받고 콘솔을 지우고 다시 print함수를 호출한다.

그리고 isFull함수를 통해 좌, 우, 상, 하에 이미 다 'a'가 채워져 있거나 배열의 밖이라면 END를 출력하고 프로그램을 종료한다.

출력화면



```
Microsoft Visual Studio 디버그 콘솔
a a 0 0 0
a a 0 0 0
a a 0 0 0
a a 0 0 0
a a 0 0 0
a a 0 0 0
Command : END
C:\Users\bjybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\Assignment_1_8\Debug\Assignment_1_8.exe(프로세스 27948
개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```



```
Microsoft Visual Studio 디버그 콘솔
a a a a a
a a a a a
a a a a a
a a a a a
a a a a a
a a a a a
Command : END
C:\Users\bjybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\Assignment_1_8\Debug\Assignment_1_8.exe(프로세스 30036
개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...
```

고찰

이번 문제도 처음에는 문제를 잘 파악하지 않고 사진만 보고 입력된 방향이 배열의 밖이거나 이미 'a'가 채워져 있는 경우에만 프로그램이 종료되는 줄 알고 문제를 구형했지만 레포트를 작성하는 도중에 다시 문제를 꼼꼼하게 읽고 코드를 수정했다. 문제를 풀 때 실행화면만을 보고 구현하는 것은 고쳐야하는 습관 중에 하나이다. 뭐든지 주어진 문제의 조건을 보고 프로그램을 구현 해야겠다. 그리고 처음에 함수에 매개변수를 전달할 때 pass by reference가 아닌 C에서 했던 것처럼 포인터 변수와 주소값을 넘겨주는 방식으로 구현을 하였는데 수업에서 배운 pass by reference 방식으로 매개변수를 성절하고 현재 x, y의 위치를 넘겨줬다.