

# 객체지향프로그래밍설계

## 2차 과제 보고서

학 과: 컴퓨터정보공학부

담당교수: 공영호

학 번: 2018202030

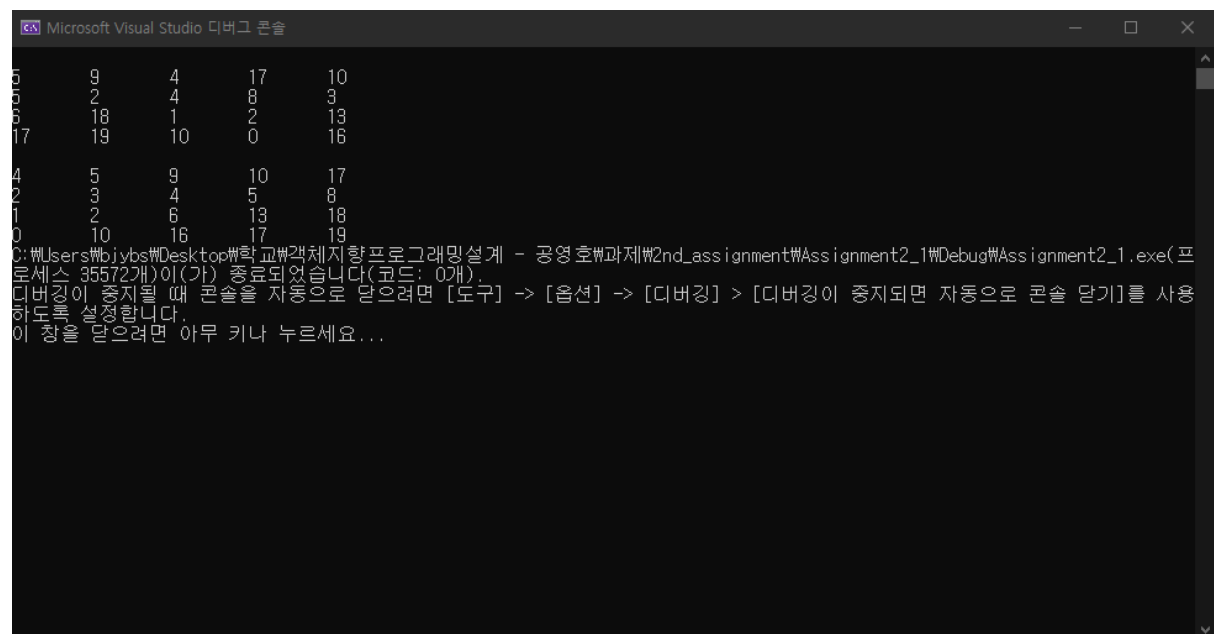
성 명: 박지용

## Assignment 2 – 1

본 문제는 0 부터 20보다 작은 숫자를 랜덤하게 4 x 5 2차원배열에 저장을 하고 각 행마다 sorting을 거치고 출력을 하는 문제이다. 문제의 핵심은 integer형 1차원 배열에 4 x 5 즉 20개의 원소를 저장하고 1차원 배열을 2차원 배열처럼 생각을 하여서 sorting을 해야한다. 또한 각 행을 sorting할 때 integer형 포인터 배열에 각 행의 첫번째 원소의 주소를 저장하여서 manipulate 해야한다.

먼저 2D array의 각 행 즉 5로 나누어 떨어지는 원소(0, 5, 10, 15)번째 들의 주소 값을 포인터 배열 row에 할당을 해주고 그 다음에 버블 소트로 각 행을 정렬해주었다.

### 결과화면



```
Microsoft Visual Studio 디버그 콘솔

5      9      4      17     10
5      2      4      8      3
5      18     1      2      13
17     19     10     0      16

4      5      9      10     17
2      3      4      5      8
1      2      6      13     18
0      10     16     17     19

C:\Users\bjybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\2nd_assignment\Assignment2_1\Debug\Assignment2_1.exe( 프로세스 35572개)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...
```

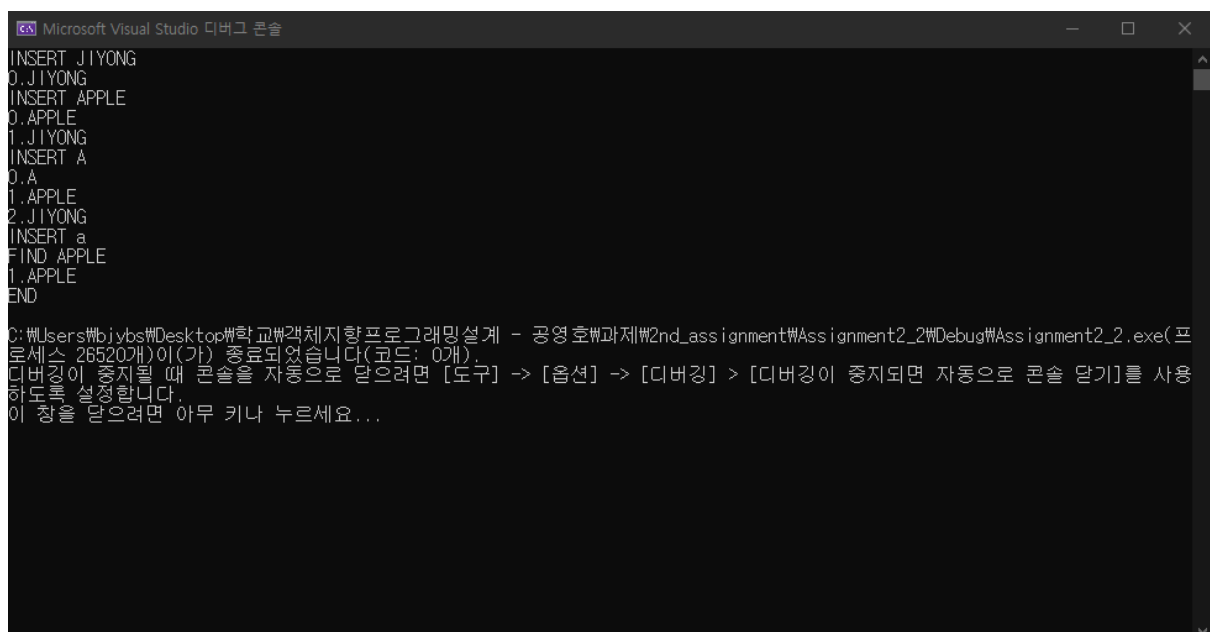
### 고찰

1차 과제에도 2차원 배열을 1차원적으로 생각을 해서 조작하는 문제를 다루어 보았기 때문에 어렵지 않게 풀 수 있었다. 처음에 2차원 배열을 이용하지 않고 문제를 풀어야 한다고 생각했을 때 쉽게 생각을 했지만 각 주소를 포인터 배열 row에 저장하고 조작을 해야 한다고 생각을 하니까 어렵다고 생각을 했는데 각 행의 첫번째 주소 값을 row에 순차적으로 저장을 해주고 sort를 진행하니 금방 문제가 해결이 되었다.

## Assignment 2 – 2

메모장을 구현하는 프로그램이다. 메모장의 정리되는 순서는 알파벳 순서로 하되 첫번째는 알파벳 순서로 나열을 하고 다음은 짧은 것 먼저 나열한다 그리고 대소문자는 구별을 안 한다. 예를 들면 이미 노트에 "JIYONG"이 있다면 다음 INSERT에 "Jiyong"이 입력이 되어도 이미 있는 단어이기 때문에 무시하게 된다.

### 출력화면



```
Microsoft Visual Studio 디버깅 콘솔
INSERT JIYONG
0. JIYONG
INSERT APPLE
0. APPLE
1. JIYONG
INSERT A
0. A
1. APPLE
2. JIYONG
INSERT a
FIND APPLE
1. APPLE
END
C:\Users\bjybs\Desktop\학 교\객체지향프로그래밍설계 - 공영호\과제\2nd_assignment\Assignment2_2\Debug\Assignment2_2.exe(프로세스 26520개)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...
```

문제를 해결하는 방안은 더블 포인터 배열을 이용해서 각 행에 입력된 문자열의 주소 값을 할당하여 스페이스를 기준으로 앞에 있는 문자열을 커맨드로 하고 뒤에 있는 문자열을 삽입하는 문자열 혹은 찾는 문자열로 처리를 해서 구현을 했다. 커맨드를 op라는 변수에 저장을 하고 switch문에서 op를 이용해서 각 INSERT FIND 함수에 각각 인가한다. INSERT함수에서 들어온 문자열과 이제까지 입력된 문자열을 비교하여 제시된 순서에 의하여 비교하는 문자의 앞 혹은 뒤로 삽입을 하게 된다. 여기서 대소문자를 구별하지 않고 "JIYONG" "Jiyong"은 같다고 생각을 해야 하기 때문에 또 하나의 더블 포인터 배열에 소문자로 바꾼 문자열들은 각각 삽입을 한다.

먼저 모든 문자가 소문자로 처리된 더블 포인터 배열에서 strcmp을 통해 대소를 구별한 뒤 해당위치에는 원래의 문자열을 삽입하는 형식으로 구현을 했다.

## 고찰

대소문자 구별을 안 하는 것을 빼고는 구현하기는 쉬웠다. 하지만 대소문자를 구별 안하고 삽입을 하는 방법을 찾기가 어려워서 시간을 많이 썼다. 내가 사용한 방법은 입력 받은 문자열에서 대문자를 다 소문자로 바꾸어서 새로운 더블 포인터에 저장을 하였고 비교 자체는 소문자로 된 배열에서 비교하고 삽입은 원래 배열에 삽입을 하는 방식으로 구현을 하였다.

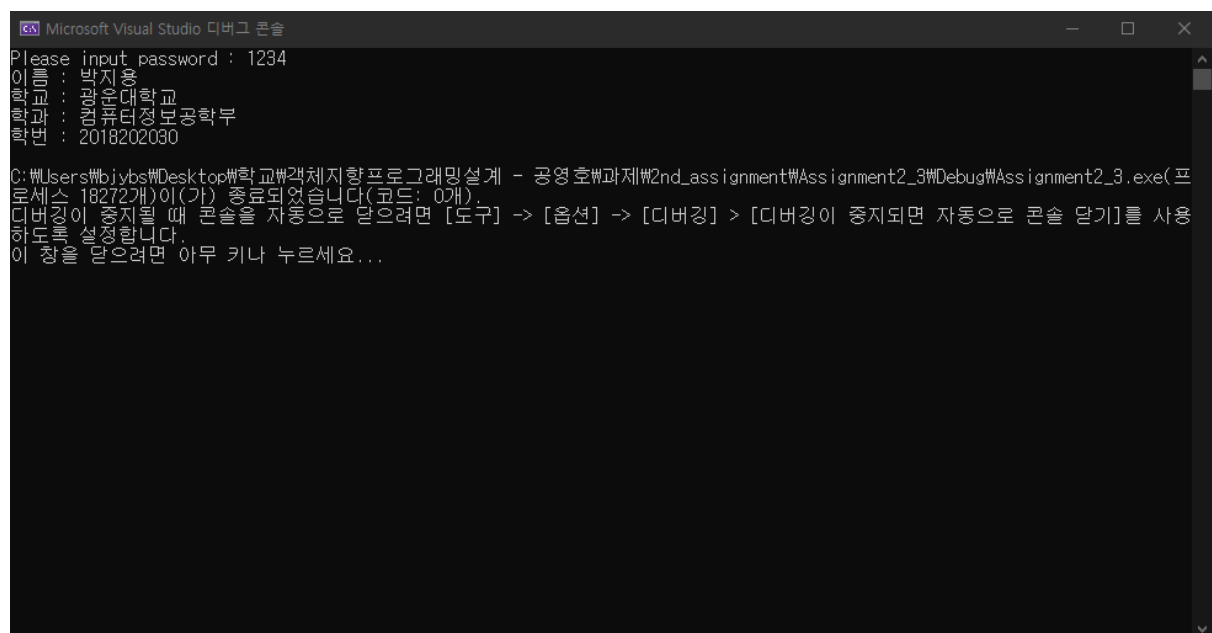
소문자로만 저장을 하기 위해서 my\_strcpy\_low라는 함수를 만들어서 문자열을 복사할 때 바로 소문자로 복사할 수 있게 구현을 하였다.

## Assignment 2 – 3

함수 포인터를 사용하는 문제이다. 함수를 선언하고 직접 사용하는 것이 아니라 주어진 void (\*fp)를 이용해서 함수의 주소 값을 저장해 간접적으로 호출하는 문제이다.

비밀번호를 입력 받아서 만약에 비밀번호가 "1234"가 맞다면 Rightprint의 주소 값을 (\*fp)에 할당하고 \*fp를 이용해서 함수 호출을 한다. 반면에 "1234"가 아니면 주소 값을 Wrongprint에 할당하고 함수를 호출한다.

## 결과화면



```
Microsoft Visual Studio 디버깅 콘솔
Please input password : 1234
이름 : 박지용
학교 : 광운대학교
학과 : 컴퓨터정보공학부
학번 : 2018202030
C:\Users\biybs\Desktop\학교\객체지향프로그래밍설계 - 공영호\과제\2nd_assignment\Assignment2_3\Debug\Assignment2_3.exe(프로세스 18272개)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...
```

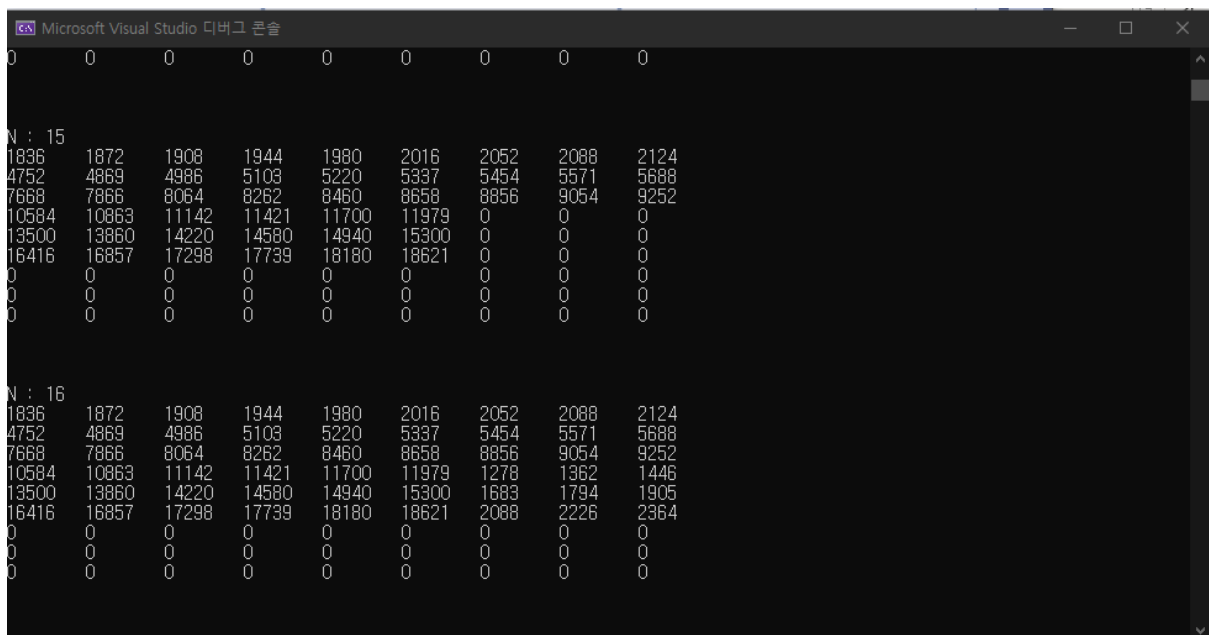
## 고찰

이 문제는 함수 포인터를 사용하는 문제이다. 함수 포인터의 개념을 알고 있으면 금방 풀 수 있는 문제이다. 일반 포인터 변수와 같이 주소 값을 함수 포인터 변수에 할당한 후에 할당된 함수 포인터를 호출해서 함수를 간접적으로 호출한다.

## Assignment 2 – 4

9 x 9배열을 3 x 3배열로 쪼개서 생각을 하고 행렬의 곱을 구현하는 문제이다. 문제에 주어진 대로 9 x 9배열을 그 자체로 행렬의 곱으로 구현하면 안 되고 3 x 3으로 쪼개어서 생각을 해야 한다.

## 결과화면



```
Microsoft Visual Studio 디버그 콘솔
0 0 0 0 0 0 0 0 0
N : 15
1836 1872 1908 1944 1980 2016 2052 2088 2124
4752 4869 4986 5103 5220 5337 5454 5571 5688
7668 7866 8064 8262 8460 8658 8856 9054 9252
10584 10863 11142 11421 11700 11979 0 0 0
13500 13860 14220 14580 14940 15300 0 0 0
16416 16857 17298 17739 18180 18621 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
N : 16
1836 1872 1908 1944 1980 2016 2052 2088 2124
4752 4869 4986 5103 5220 5337 5454 5571 5688
7668 7866 8064 8262 8460 8658 8856 9054 9252
10584 10863 11142 11421 11700 11979 1278 1362 1446
13500 13860 14220 14580 14940 15300 1683 1794 1905
16416 16857 17298 17739 18180 18621 2088 2226 2364
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
```

## 고찰

이 문제는 제일 난해한 문제였다. 보기에는 행렬의 곱을 이용해서 쉽게 풀릴 것 같았지만 가장 어려운 문제였다. 내가 한 방법은 먼저 9 x 9 2차원 배열에 1차원 배열로 생각했을 때의

원소 번호로 초기화는 시켰다. 그리고 3 x 3 행렬을 곱하는 mulMatrix함수를 만들었다. 이 함수의 매개변수는 9 x 9 곱이 행해지는 배열과 출력에 사용되고 3 x 3배열의 곱이 저장되는 배열과 출력에 사용이 되는 배열의 position을 위치하는 변수와 9 x 9에서 3 x 3행렬이 곱해질 때 각 3 x 3행렬의 좌 상단 원소 즉, 1차원으로 생각했을때의 원소를 전달해준다.

```

Microsoft Visual Studio 디버그 콘솔
N : 1
miniMatrix[i][j] : 0 0*0
miniMatrix[i][j] : 9 1*9
miniMatrix[i][j] : 45 2*18
miniMatrix[i][j] : 0 0*1
miniMatrix[i][j] : 10 1*10
miniMatrix[i][j] : 48 2*19
miniMatrix[i][j] : 0 0*2
miniMatrix[i][j] : 11 1*11
miniMatrix[i][j] : 51 2*20
miniMatrix[i][j] : 0 9*0
miniMatrix[i][j] : 90 10*9
miniMatrix[i][j] : 288 11*18
miniMatrix[i][j] : 9 9*1
miniMatrix[i][j] : 109 10*10
miniMatrix[i][j] : 318 11*19
miniMatrix[i][j] : 18 9*2
miniMatrix[i][j] : 128 10*11
miniMatrix[i][j] : 348 11*20
miniMatrix[i][j] : 0 18*0
miniMatrix[i][j] : 171 19*9
miniMatrix[i][j] : 531 20*18
miniMatrix[i][j] : 18 18*1
miniMatrix[i][j] : 208 19*10
miniMatrix[i][j] : 588 20*19
miniMatrix[i][j] : 36 18*2
miniMatrix[i][j] : 245 19*11
miniMatrix[i][j] : 645 20*20
45 48 51 0 0 0 0 0
288 318 348 0 0 0 0 0

```

위 사진은 3x3이 곱해지는 것을 출력한 사진이다. 9 x 9 배열을 3 x 3배열로 생각을 하면 3 x 3행렬은 곱하기가 27번이 요구된다. 위 사진과 같이 9 x 9 행렬에서 1 행 1열의 3 x 3결과 1 행 1 열 3 x 3행렬을 곱한 식이다. 그 다음은 1 행 2 열과 2 행 1열 행렬 곱을 더해주고 1행 3열과 3행 1열의 행렬 곱을 더해주면 9 x 9 행렬에서 1 행 1 열 3 x 3행렬이 완성이 된다.

```
Microsoft Visual Studio 디버그 콘솔
miniMatrix[i][j] : 225 4*36
miniMatrix[i][j] : 450 5*45
miniMatrix[i][j] : 84 3*28
miniMatrix[i][j] : 232 4*37
miniMatrix[i][j] : 462 5*46
miniMatrix[i][j] : 87 3*29
miniMatrix[i][j] : 239 4*38
miniMatrix[i][j] : 474 5*47
miniMatrix[i][j] : 324 12*27
miniMatrix[i][j] : 792 13*36
miniMatrix[i][j] : 1422 14*45
miniMatrix[i][j] : 336 12*28
miniMatrix[i][j] : 817 13*37
miniMatrix[i][j] : 1461 14*46
miniMatrix[i][j] : 348 12*29
miniMatrix[i][j] : 842 13*38
miniMatrix[i][j] : 1500 14*47
miniMatrix[i][j] : 567 21*27
miniMatrix[i][j] : 1359 22*36
miniMatrix[i][j] : 2394 23*45
miniMatrix[i][j] : 588 21*28
miniMatrix[i][j] : 1402 22*37
miniMatrix[i][j] : 2460 23*46
miniMatrix[i][j] : 609 21*29
miniMatrix[i][j] : 1445 22*38
miniMatrix[i][j] : 2526 23*47
495 510 525 0 0 0 0 0
1710 1779 1848 0 0 0 0 0
2925 3048 3171 0 0 0 0 0
0 0 0 0 0 0 0 0
```

```
Microsoft Visual Studio 디버그 콘솔
miniMatrix[i][j] : 324 6*54
miniMatrix[i][j] : 765 7*63
miniMatrix[i][j] : 1341 8*72
miniMatrix[i][j] : 330 6*55
miniMatrix[i][j] : 778 7*64
miniMatrix[i][j] : 1362 8*73
miniMatrix[i][j] : 336 6*56
miniMatrix[i][j] : 791 7*65
miniMatrix[i][j] : 1383 8*74
miniMatrix[i][j] : 810 15*54
miniMatrix[i][j] : 1818 16*63
miniMatrix[i][j] : 3042 17*72
miniMatrix[i][j] : 825 15*55
miniMatrix[i][j] : 1849 16*64
miniMatrix[i][j] : 3090 17*73
miniMatrix[i][j] : 840 15*56
miniMatrix[i][j] : 1880 16*65
miniMatrix[i][j] : 3138 17*74
miniMatrix[i][j] : 1296 24*54
miniMatrix[i][j] : 2871 25*63
miniMatrix[i][j] : 4743 26*72
miniMatrix[i][j] : 1320 24*55
miniMatrix[i][j] : 2920 25*64
miniMatrix[i][j] : 4818 26*73
miniMatrix[i][j] : 1344 24*56
miniMatrix[i][j] : 2969 25*65
miniMatrix[i][j] : 4893 26*74
1836 1872 1908 0 0 0 0 0
4752 4869 4986 0 0 0 0 0
7668 7866 8064 0 0 0 0 0
```

## 고찰

코드를 보다시피 내가 본인이 봤을 때도 비효율적이라고 생각을 했다. 코드를 최대한 줄일 수 있는 선까지 줄여보았지만 이게 최선이였다. 처음부터 3 x 3행렬을 곱하는 방법을 생각했지만 생각보다 어려웠고 반복문이 중첩이 되다 보니 어떻게 돌아가는지 쉽게 파악이 안되었다. 결국엔 반복문을 이용하여 각각의 행렬의 좌상단의 원소를 매개변수로 넘겨주고 그 받은 매개 변수로 3 x 3행렬곱을 하는 방식으로 구현을 하였다.

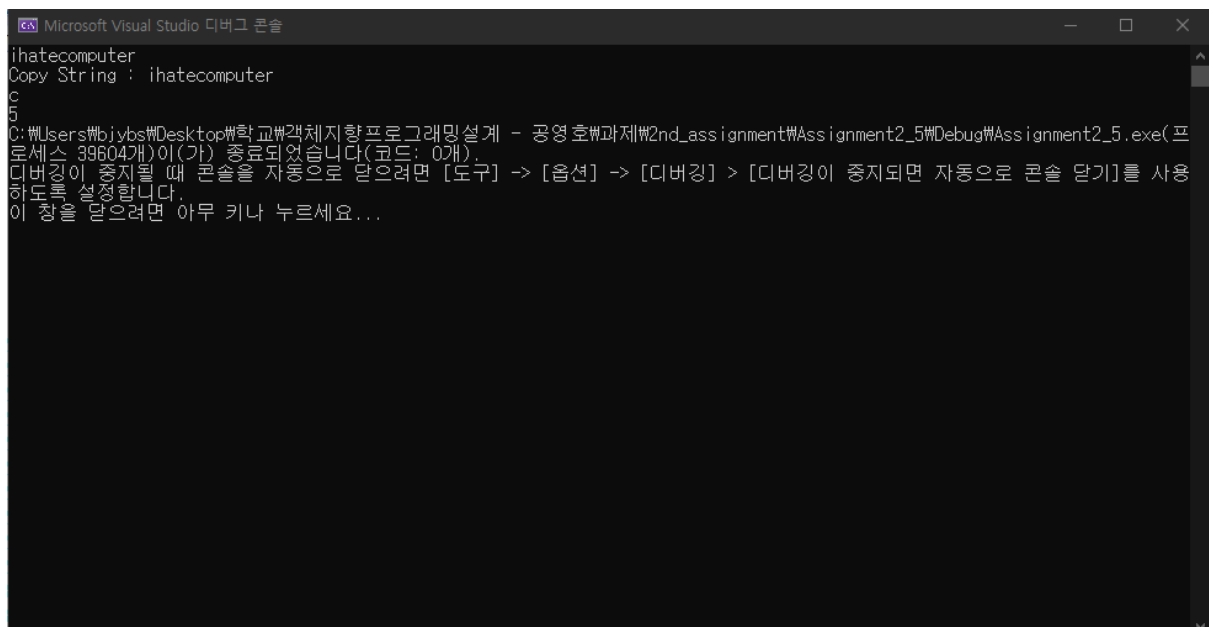
제일 오래 걸린 부분은 의외로 3 x 3 행렬의 곱 이였다. 9 x 9 행렬 곱을 구현할 생각만

해서 그런지 계속 최종 값이 다르게 나와 확인을 해보니 N의 단계가 29이 아닌 9로 나와있었고 3 x 3행렬에서 1 행 1열의 곱만 구현을 하고 나머지는 구현을 안 한 것이 문제였다.

## Assignment 2 – 5

클래스를 이용한 기초 문제이다. 클래스를 이용하여서 문자열을 입력 받고 입력 받은 문자열에서 특정 문자를 입력 받으면 해당되는 문자를 찾아서 해당문자의 원소 위치를 출력해주는 클래스를 만드는 문제이다.

### 결과화면



```
Microsoft Visual Studio 디버그 콘솔
ihatecomputer
Copy String : ihatecomputer
C
5
C:\Users\wbjybs\Desktop\학 교\객체지향프로그래밍설계 - 공영호\과제\2nd_assignment\Assignment2_5\Debug\Assignment2_5.exe( 프로세스 39604개)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...
```

클래스의 private은 char \* 캐릭터형 포인터가 존재하고 그 포인터에 동적할당을 하여 문자열을 저장하고 조작하는 문제이다. Private에 속한 변수는 클래스의 외부에서 접근이 불가하기 때문에 클래스 안에서 초기화를 해주어야 한다. 그러기 위해서는 public에 strcpy함수를 구현하고 public에서 문자열을 매개변수로 전달받은 만큼 복사를 한다. 또한 find\_char함수를 통해 문자열을 문자 단위로 널문자를 만나기 전까지 검사를 하고 매개변수로 입력 받은 문자와 같으면 문자의 위치를 출력하고 만약에 문자가 없다면 -1을 출력한다.



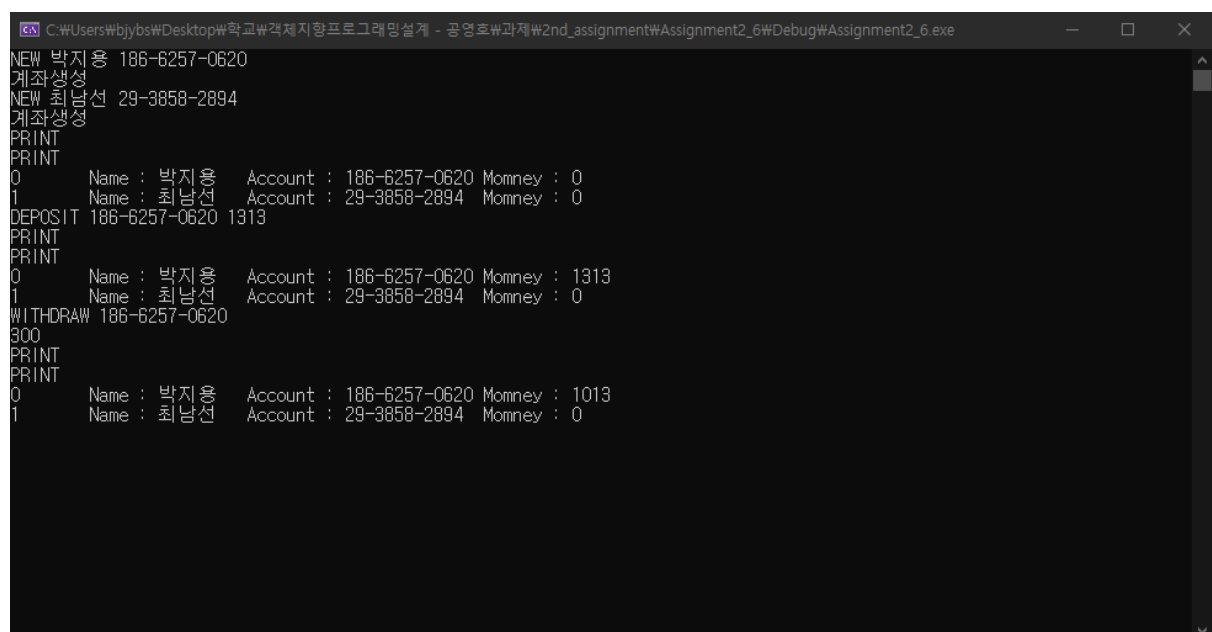
## 고찰

간단한 클래스를 이용한 문제지만 소멸자를 이용해 동적할당을 해제하는 것이 생각보다 어려웠다. 생성자와 소멸자를 사용해 본적이 없기 때문에 먼저 생성자와 소멸자 클래스 특성들을 익히는 데에 중점을 두고 공부를 진행하고 코드화 시켰다.

## Assignment 2 – 6

클래스를 이용한 문제이고 5번과 다르게 함수로 private변수를 조작하는 문자가 아닌 생성자를 이용해서 private에 있는 변수들을 초기화하는 문제이다. 또한 객체를 생성할 때 포인터 배열로 선언을 한 후에 커맨드가 계좌 생성 커맨드 일 때 생성자를 호출하여 배열에 동적 생성한다.

## 결과화면



```
C:\Users\Wbjybs\Desktop\학교\객체지향프로그래밍실계 - 공영호\과제\2nd_assignment\Assignment2_6\Debug\Assignment2_6.exe
NEW 박지용 186-6257-0620
계좌생성
NEW 최남선 29-3858-2894
계좌생성
PRINT
PRINT
0      Name : 박지용      Account : 186-6257-0620 Momney : 0
1      Name : 최남선      Account : 29-3858-2894 Momney : 0
DEPOSIT 186-6257-0620 1313
PRINT
PRINT
0      Name : 박지용      Account : 186-6257-0620 Momney : 1313
1      Name : 최남선      Account : 29-3858-2894 Momney : 0
WITHDRAW 186-6257-0620
300
PRINT
PRINT
0      Name : 박지용      Account : 186-6257-0620 Momney : 1013
1      Name : 최남선      Account : 29-3858-2894 Momney : 0
```

처음에 문제를 풀고나서 다음문제로 넘어가고 문제 검토를 할 때 처음 구현을 할 때 단순히 객체배열로 구현을 해서 다시 구현을 하게 되었다. 생성자를 통해서 private 변수를 초기화 시키고 오버로딩을 통해 계좌를 생성할 때 동적으로 오버로딩 생성자를 호출하여 객체를 생성했다. 각 커맨드에 대응되는 함수를 public에 생성하여 입력된 계좌번호와 해당 객체가 저장한 계좌번호가 일치하게 되면 인출과 저축을 할 수 있게 구현하였다.

마지막으로는 소멸자를 통해 각 객체에 동적할당이 된 Name과 Account\_number를 동적할당 해제시켜 주었고 계좌생성에 따른 객체의 동적 생성도 delete[]함수를 통해 할당 해제해주었다.

## 고찰

객체를 동적으로 생성하는 것이 어색하고 처음 해보는 것이라 이해하기가 힘들었다. 또한 생성자 오버로딩을 통해서 동적 생성을 하는게 생성자와 소멸자가 어떻게 어떤 시기에 호출되는지 한 눈에 확인하기가 어려워서 각 생성자와 소멸자에 cout 함수를 통해서 "생성자" "소멸자" 문구를 통해서 익혔다.

. 참조와 -> 참조도 어떨 때 쓰는지 잘 기억이 잘 안 나서 찾아본 결과 점 연산자 앞에 변수나 포인터가 아닌 저장하려는 데이터가 있을 때 사용하고 ->포인터로 요소를 선택하고 앞에 포인터가 있는 경우에 사용한다.

## Assignment 2 – 7

클래스의 부모와 자식을 활용하는 문제이다. 자식 클래스를 생성하여 부모의 private변수를 접근하는 문제이다. 부모 클래스는 대학 클래스이고 그의 자식 클래스는 컴퓨터정보공학부 클래스이다. 컴퓨터정보공학부의 자식 클래스는 학생정보 클래스이다. 커맨드가 INSERT이면 각 해당되는 정보를 입력 받은 후 학생정보 생성자를 통해서 모든 정보를 넘겨준다. 제일 작은 자식 클래스부터 제일 부모 클래스인 대학 클래스에 이르기까지 넘겨받은 클래스를 각 클래스 private 해당되는 변수들에 저장한다.

## 결과화면

```
Microsoft Visual Studio 디버그 콘솔
INSERT
학교 : 광운대학교
학번 : 2018202030
학년 : 2
객체지향 프로그래밍 학점 : A
디지털 논리회로 학점 : A
이름 : 박지웅
나이 : 24
INSERT
학교 : 하버드대학교
학번 : 2018303020
학년 : 2
객체지향 프로그래밍 학점 : C
디지털 논리회로 학점 : C
이름 : 박초코
나이 : 8
FIND
이름 : 박초코
학번 : 2018303020
학교 : 박초코
학번 : 2018303020
학년 : 2
객체지향 프로그래밍 학점 : C
디지털 논리회로 학점 : C
EXIT
C:\Users\bjybs\Desktop\학교\객체지향 프로그래밍 설계 - 공영호\과제\2nd_assignment\Assignment2_7\Debug\Assignment2_7.exe( 프로세스 9608개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
```

학생정보는 100개까지 받을 수 있고 커맨드가 INSERT이면 count를 늘리며 학생정보의 수는 카운트한다. 커맨드가 INSERT이면 학교, 학번, 두 수업의 학점, 이름과 나이를 차례대로 입력 받고 카운트하는 count에 해당하는 stu 객체를 동적 생성한다. `Stu[count] = new 학생정보()`.

## 고찰

상속에 개념에 대해서 이해하기가 조금 어려웠다. 상속 관계에 있을 때 객체를 생성하면 맨 위 부모 클래스인 대학 클래스의 생성자부터 차례대로 생성자가 호출이 된다는 사실을 이해하기가 어려웠다. 자식 클래스 생성자에서 모든 필요한 매개변수들을 받아서 위 부모 클래스에게 넘겨주면서 각 private에 있는 변수들을 초기화 해준다. 아직 개념이 애매모호 하지만 계속 클래스를 사용해 보면서 개념들을 그때 그때 확인하면서 숙지를 해야할 것 같다.

## Assignment 2 – 8

동물들의 달리기 시합을 하는 문제이다. 모든 동물이 arrival에 도착을 해야 프로그램이 종료되며 달리기 시합은 먹이사슬에 기반하며 만약에 먹이사슬에서 상위인 동물을 하위인 동물이 만나면 동물은 상위 포식자에게 잡아 먹히고 다시 출발선부터 시작하여 달리게 시작된다. 각 동물의 객체를 만들어서 foodchain과 한 번에 얼마큼 움직이는지를 나타내는 distance와 arrival을 이용해서 1차원 배열을 이용해 움직이게 한다. 입력 받은 알파벳 조합의 순서로 동물들이 출발하

고 매 움직임마다 출력을 하여 달리기 상태를 보여준다.

## 결과화면

사용자가 입력한 시퀀스대로 동물들이 출발하게 만든다. 예를 들어 위 출력화면에 입력된 시퀀스는 S F T E이므로 첫번째로 뱀이 출발하고 다음 턴에 개구리 호랑이 독수리가 차례대로 출발한다 제일 먼저 출발한 뱀은 2턴 뒤에 출발한 호랑이와 같은 위치에 놓이고 뱀은 호랑이에게 잡아 먹혀 다시 출발점으로 돌아가게 된다.

출발 순서를 정해주는 방법은 시퀀스를 입력을 받을 때 각 동물에 해당되는 변수를 따로 만들어 몇 번째로 입력이 되었는지 저장한다. 예를 들어 S F T E라고 하면 S = 0, F = 1, T = 2, E = 3으로 저장이 되고 while루프에서 모든 동물들의 변수를 하나씩 줄여 0보다 작거나 같으면 출발하도록 만들었다.

각 동물들 클래스를 하나의 부모 클래스의 상속되게 만들었고 각 동물들의 객체를 생성할 때 클래스에 정의된 오버로딩이 된 생성자를 이용해 동적으로 생성하게 만들었다. 해당 생성자는 foodchain을 매개변수로 받고 이 foodchain이 동물과 동물이 같은 위치에 속해 있을 때 foodchain이 큰 동물은 그대로 남고 작은 동물은 원점으로 돌아가게 하기 위한 변수이다.

부모 클래스에 있는 virtual void run() 매 사이클마다 동물들이 distance만큼 앞으로 전진하게 하는 함수이다. 만약에 상속된 자식 클래스에서 run()함수를 호출하면 부모 클래스에 있는 run()함수를 호출하기 때문에 상속된 동물 클래스에서 각 각 고유의 distance로 전진하기 위해서는 virtual void run()함수를 각 동물 클래스마다 정의를 해주어야 각 동물 객체에서 run()함수를 호

호 하였을 때 각 동물 클래스의 run()함수가 실행이 되게 된다.

## 고찰

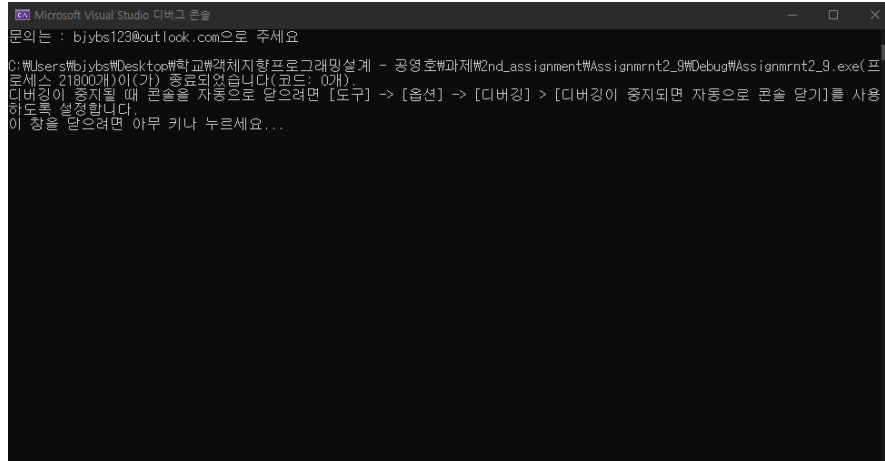
이번 문제는 사진으로 봤을 때 되게 어렵게 보여서 거부감이 들었다. 먼저 virtual 함수에 대해서 알지를 못했기 때문에 인터넷으로 해당 함수가 무슨 작용을 하는지 알아봐야 했고 또한 virtual 함수를 안 쓰고 각 클래스에 동일한 동작을 하는 함수를 생성하면 왜 안 되는지 꼭 virtual 함수를 써야 하는지를 고민했다.

또한 동물들이 정해진 시퀀스 대로 출발을 해야 하기 때문에 어떻게 하나의 반복문을 이용해서 동물들이 각 각 출발하게 할지 한번에 생각이 안 떠올랐다. 내가 사용한 방법을 각 동물들에 대응이 되는 변수를 만들어서 시퀀스에서 먼저 나온 동물의 수서대로 integer형 변수에 숫자를 넣어 반복문에서 감소시키며 0보다 같거나 작으면 동물들의 run()함수가 작동하게 하는 방법을 사용했다. 뷰모 클래스에 있는 private 변수를 초기화 혹은 값을 저장할 수 없기 때문에 Set과 Get함수를 적절히 선언해 주어서 부모 클래스에 이는 값들은 초기화하고 불러올 수 있었다.

## Assignment 2 – 9

Mystring 클래스를 만들어 cout함수를 구현하는 문제이다. 객체를 만들고 문자열을 넘겨주면 생성자를 통해서 문자열을 저장하고 두 객체를 더해서 cout을 하게 되면 strcat처럼 작용을 하여 두 문자열이 연결되어 출력하게 된다. 원래 cout은 객체를 출력할 수 없기 때문에 객체를 출력할 수 있게 처리를 해야 한다. 그러기 위해서는 ostream 내부에 객체를 받는 멤버함수로 구현할 수 없기 때문에 전역변수로 구현을 해주어야 한다.

## 결과화면



문제에 명시되어 있는 것처럼 구현을 하기위해선 제일 먼저 Mystring이라는 클래스는 만들어야 한다. 그 다음으로 문자열을 매개변수로 받는 생성자를 만들어야 하고 입력 받은 문자열을 private변수에 char\* 변수에 저장을 한다. 그리고 객체와 객체를 더해주는 + 연산자에 대한 오버로드 함수를 구현해 주어야 하고 그 더해진 객체들을 새로운 객체에 넣기 위해서는 = 연산자에 대해서 오버로드를 해주어야 한다. + 연산자를 통해서 뒤에 나온 문자열을 my로 받고 + 연산자 앞에 있는 문자열은 str로 조작이 가능하다. 두 문자열의 길이를 구하고 합산하여 임시 문자열을 만들어 저장을 한다. 그 임시 문자열을 str에 복사를 한 후 할당을 해제하고 해당 클래스 자체 문자열의 주소를 를 반환한다. 그 다음 = 연산자는 받은 클래스의 자체의 값을 이용해 자신의 문자열에 들어온 문자열을 저장하고 문자열의 주소를 반환한다. 결론적으로 전역함수로 선언된 << 연산자의 오버로딩 함수는 객체를 받고 출력을 하게 된다.

## 고찰

처음에 friend와 연산자 자체를 오버로드 해서 사용을 한 다는게 감이 장 안 왔다. 처음엔 쉽게 변수 + 변수를 생각해서 쉽게 풀린 것 같다고 생각을 했지만 객체 + 객체 라는 것을 생각을 못 하고 쉽게 생각을 해서 더 어렵게 느껴진 것 같다. 인터넷에서 연산자의 오버로드에 대해서 공부를 했고 그 중에 <<연산자는 c++ 표준에 정의 되어있기 때문에 ostream에 자체 적으로 함수를 구현이 어렵기 때문에 전역 함수 구현을 하고 클래스에 friend로 연결을 하여서 구현을 하였다.