

## 과제: 객체지향프로그래밍실습 (목 1,2) Assignment 1-2

학부: 컴퓨터정보공학부

학번: 2019202021

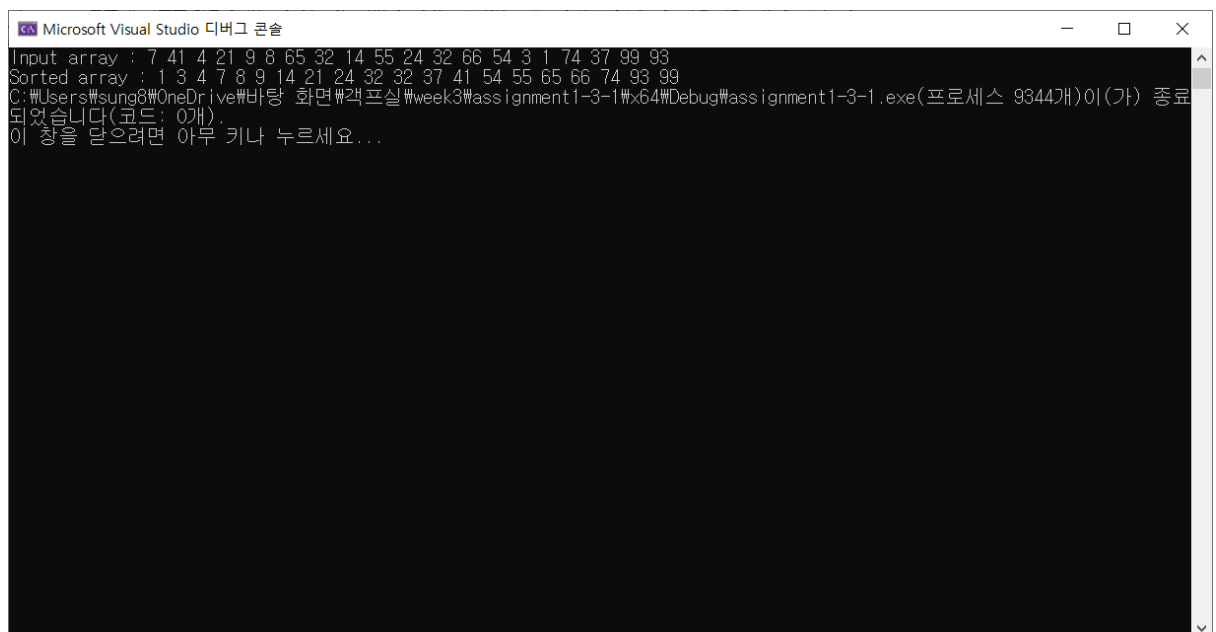
이름 정성엽

### <1번 문제>

#### 1) 문제 설명

- txt 파일로부터 임의의 숫자 20개를 입력 받아 ascending order로 정렬해 출력하는 프로그램을 구현하는 것으로 STL이나 그 외 Built in function을 이용하지 않고 정렬을 하도록 한다. 그러므로 tmp의 임의의 공간을 변수로 만들어 자리를 바꾸도록 한다. fin 함수를 이용할 때 공백을 기준으로 끊어서 읽음을 알고 이용하여 배열에 저장한다.

#### 2) 결과 화면



```
Microsoft Visual Studio 디버그 콘솔
Input array : 7 41 4 21 9 8 65 32 14 55 24 32 66 54 3 1 74 37 99 93
Sorted array : 1 3 4 7 8 9 14 21 24 32 32 37 41 54 55 65 66 74 93 99
C:\Users\#sung8#OneDrive#바탕 화면#객 프 실#week3#assignment1-3-1#x64#Debug#assignment1-3-1.exe (프로세스 9344개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

예시와 같은 수가 들어간 파일을 열었을 때 결과.

#### 3) 고찰

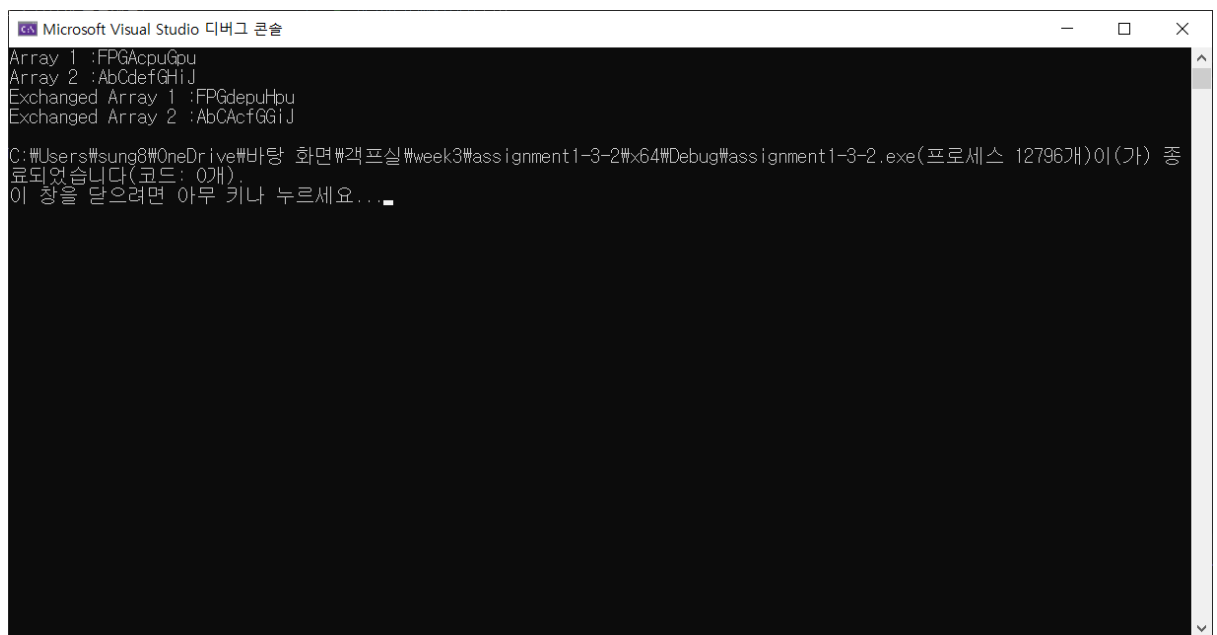
- 파일을 열고 프로그램을 마무리 또는 파일을 더 이상 사용하지 않는 경우 close()를 이용하여 파일을 닫도록 해야한다. 오름차순으로 정렬할 때 전체를 다 확인한 후 더 작은 값만 확인 후 바꾸는 방법도 있지만 현재 수보다 작은 경우 바로 바꾸면 변수 하나를 덜 선언해도 된다고 생각하여 바로 바뀌서 진행하도록 프로그램을 만들었다.

## <2번 문제>

### 1) 문제 설명

- 같은 길이의 2개의 영어 문자열을 입력 받아 비교해서 문자를 이동하는 프로그램을 구현하는 것으로, 대·소문자 구별 없이 문자의 순서에 맞게 뒤에 있는 문자는 첫 번째 문자열로 앞에 있는 문자는 두 번째 문자열로 이동하게 한다. 이를 비교 하기 위해서는 각 문자의 아스키코드가 있음을 알고 이를 비교하여 이동시킨다.

### 2) 결과 화면



```
Microsoft Visual Studio 디버그 콘솔
Array 1 :FPGAcpuGpu
Array 2 :AbCdefGHij
Exchanged Array 1 :FPGdepuHpu
Exchanged Array 2 :AbCAcfGGij
C:\Users\#sung8\OneDrive\바탕 화면\2객 프실\week3\assignment1-3-2\#x64\Debug\assignment1-3-2.exe(프로세스 12796개)이(가) 종
료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요....
```

FPGAcpuGpu, AbCdefGHij를 각각 입력했을 때 결과.

### 3) 고찰

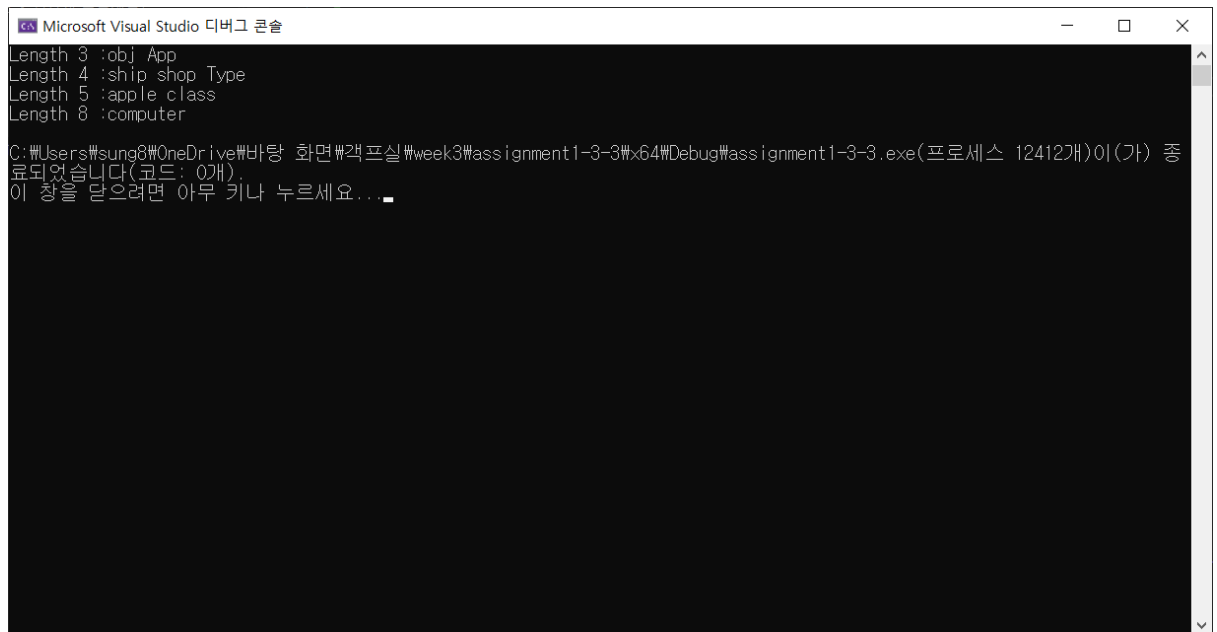
- 대·소문자는 같은 알파벳인 경우 아스키코드에서 32만큼의 차이가 있으므로 대·소문자 구분 없이 비교하기 위해선 같은 선상에 놓을 필요가 있다고 판단하여 둘다 소문자일 때, 첫 번째만 대문자일때, 두 번째만 대문자일때, 모두 대문자일 때를 switch를 이용해 경우를 나누어 비교하게 진행했다. 자리를 바꾸는 경우 1번 문제와 마찬가지로 tmp라는 임의의 변수를 이용해 자리를 바꾸도록 했다.

### <3번 문제>

#### 1) 문제 설명

- txt 파일로부터 임의의 길이 최대 20인 영단어, 최대 100개를 입력 받아서 영단어 길이가 별로 구분하여 정리 후 출력하도록 한다. 문자열 길이는 문자열 길이를 구하는 함수를 직접 구현해서 구한다. 각 길이 별로 단어의 개수가 다르고 단어가 있는 경우에만 출력하도록 한다.

#### 2) 결과 화면



```
Microsoft Visual Studio 디버그 콘솔
Length 3 :obj App
Length 4 :ship shop Type
Length 5 :apple class
Length 8 :computer

C:\Users\#sung8#OneDrive#바탕 화면#객 프 실#week3#assignment1-3-3#x64#Debug#assignment1-3-3.exe(프로세스 12412개)이 (가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요....
```

예시와 같은 파일을 이용했을 때 결과

#### 3) 고찰

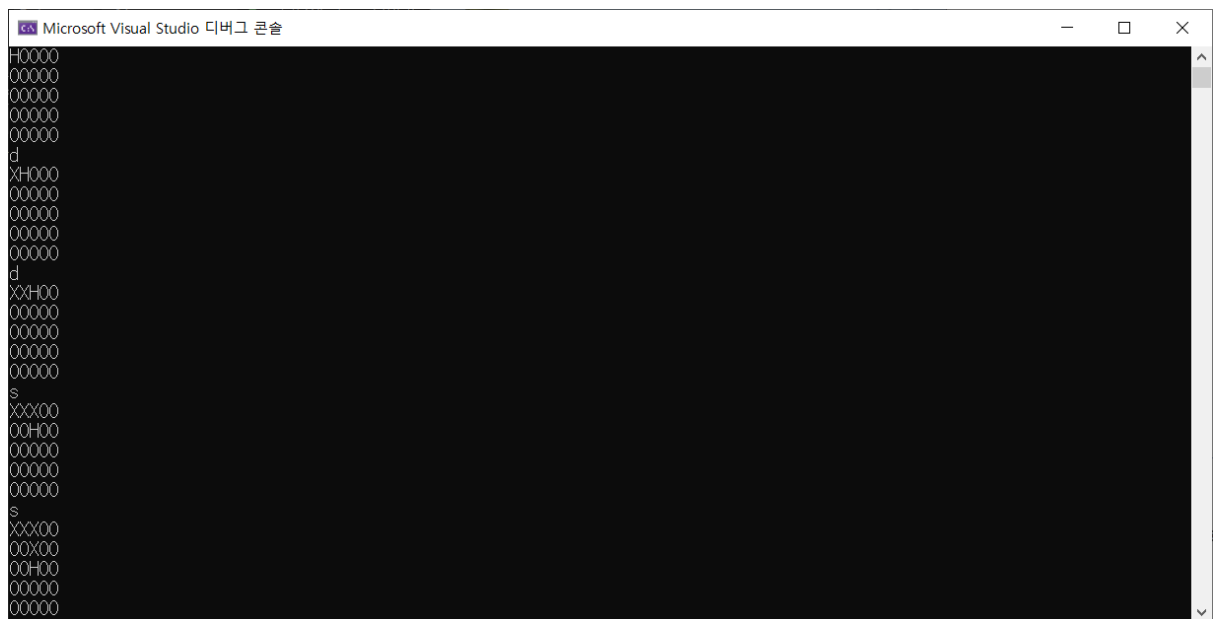
- 단어의 길이는 최대 20이고 개수는 100개이므로 2차원 배열을 이용하여 각 문자를 저장하였다. 그리고 이 단어들을 길이를 널문자를 기준으로 길이를 구분하였다. 이 길이의 값을 배열에 저장하여 각 단어별로 길이를 저장하였고 각 길이가 있을 때마다 길이 개수 관련 배열을 1씩 증가시켰고 후에 길이 별 단어 개수가 1이상인 경우에만 `cout << Length (num)`이 출력되도록 하고 해당 길이를 가진 단어를 출력하도록 프로그램을 만들었다.

#### <4번 문제>

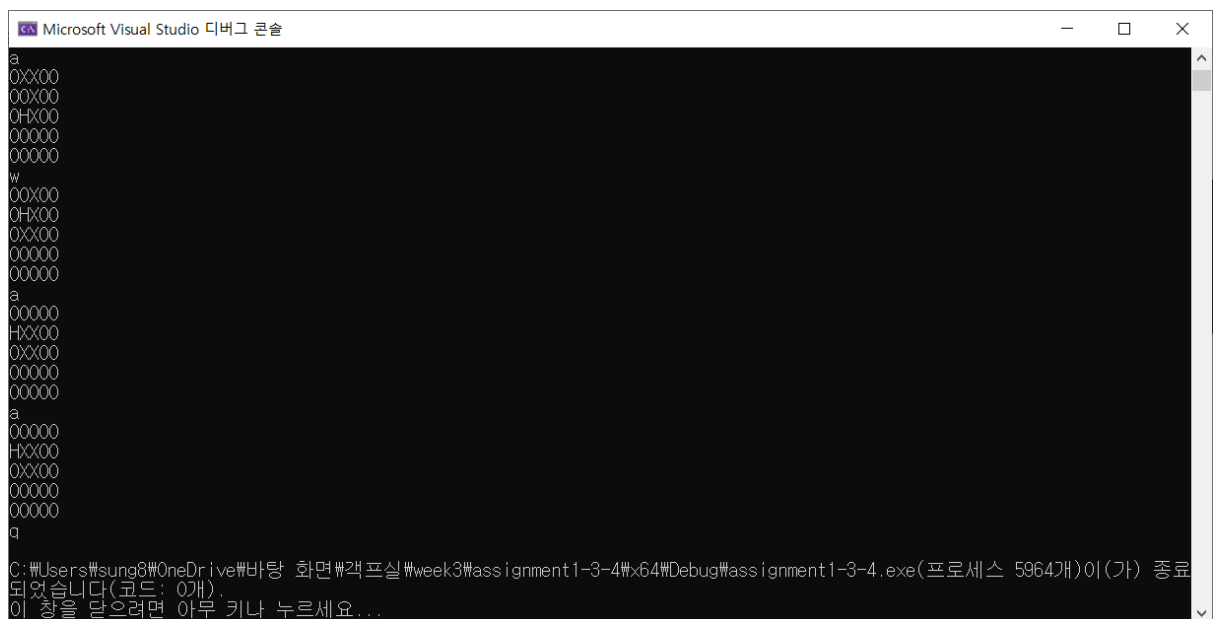
##### 1) 문제 설명

- 5x5보드에서 command를 입력 받아 노드를 움직이는 프로그램으로 이전 과제 1-2의 3번 문제를 변형시켜 완성하되 노드가 움직이기 이전 4번째 경로는 'X'로 표현하고 나머지는 'O'로 표현한다. 노드는 좌상단 끝(0,0)에서 시작하며 'w', 'a', 's', 'd', 'q'의 명령만 듣는다. 위치를 표현하기 위해 2차원 배열을 이용한다.

##### 2) 결과 화면



```
Microsoft Visual Studio 디버거 콘솔
H0000
00000
00000
00000
00000
00000
d
X0000
00000
00000
00000
00000
d
X0000
00000
00000
00000
s
X0000
00000
00000
00000
s
X0000
00000
00000
00000
s
X0000
00000
00000
00000
```



```
Microsoft Visual Studio 디버거 콘솔
a
00000
00000
00000
00000
00000
w
00000
00000
00000
00000
00000
a
00000
00000
00000
00000
a
00000
00000
00000
00000
q
C:\Users\sung8\OneDrive\바탕 화면\객 프 실\week3\assignment1-3-4\Debug\assignment1-3-4.exe(프로세스 5964개)이(가) 종료
되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

예시와 같이 이동했을 때의 결과.

### 3) 고찰

- 이전에 노드가 있었던 경로를 저장하기 위해 `pastX[4]`, `pastY[4]` 배열을 `int`형으로 선언하여 이전 위치를 저장하였고 이동할 때마다 이전 기록은 뒤로 밀어 저장하였다. 4개 이상이 된 경우에는 마지막 경로 기록은 'O'로 선언해주고 미룬 다음 새로운 기록을 받게 한다.