**학과: 게임공학과**

**학번: 2019180013**

**이름: 두혁찬**

Mail: dhc518@tukorea.ac.kr

Phone: 010-4051-4840

Github:

**#2-1. 아래의 출력 결과와 같이 두 사람이 하는 가위, 바위, 보 게임을 만들어보자. “가위”, “바위”, “보” 중 하나의 문자열을 입력받고, 누가 이겼는지 판단하여 승자를 출력한다. (비기게 되면 “비겼습니다” 라고 출력해야 한다.)**

**<실행결과>**

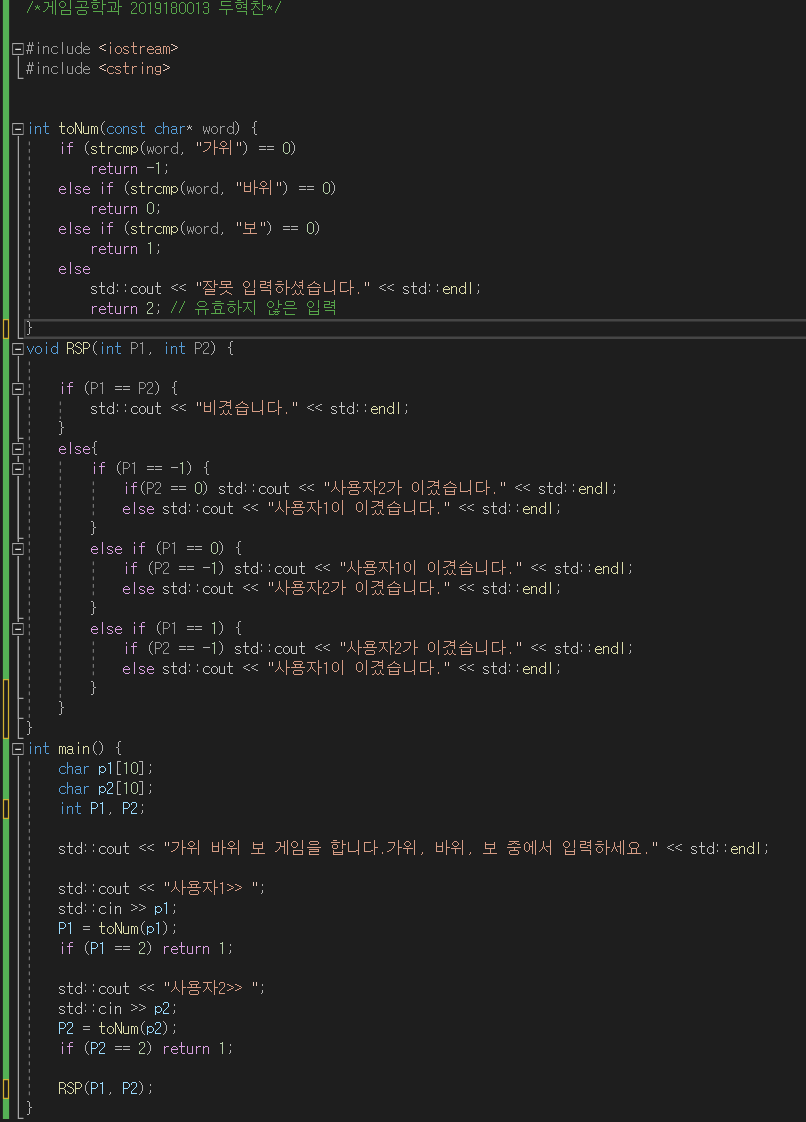
가위 바위 보 게임을 합니다. 가위, 바위, 보 중에서 입력하세요.

사용자1>> 가위

사용자2>> 바위

사용자 2가 이겼습니다.

* 소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)



* 출력화면 캡쳐

텍스트, 스크린샷, 디스플레이, 폰트이(가) 표시된 사진

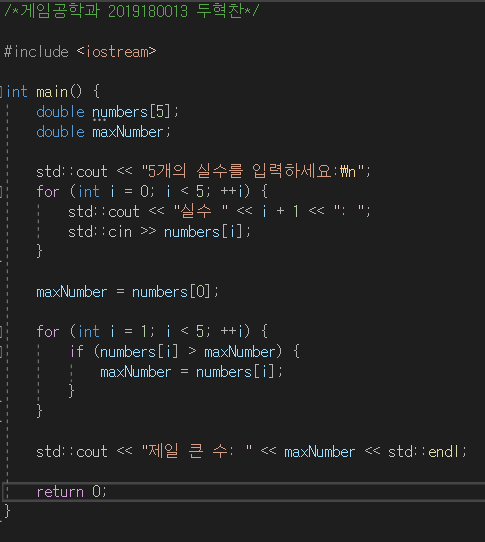
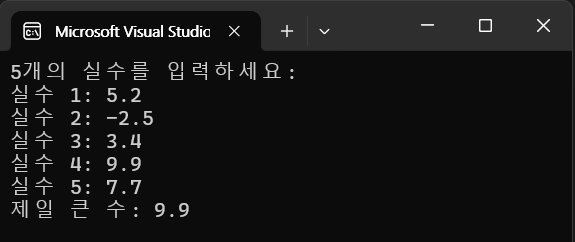
자동 생성된 설명

**#2-2. 소수점을 가지는 5개의 실수를 입력 받아 제일 큰 수를 화면에 출력하라.**

**<실행결과>**

5개의 실수를 입력하시오>> 5.2, -2.5, 3.4, 9.9, 7.7

제일 큰 수 = 9.9

* 소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)
* 
* 출력화면 캡쳐
* 

**#2-3. 다음과 같이 “yes”가 입력될 때까지 종료하지 않는 프로그램을 작성하라. 사용자로부터의 입력은 cin.getlin() 함수를 사용하라**

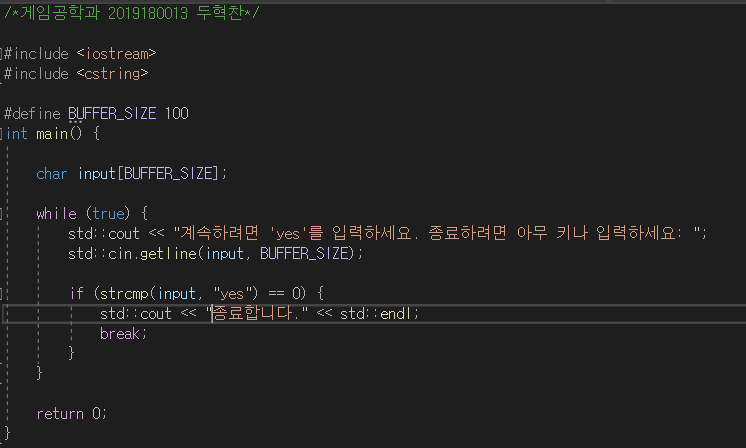
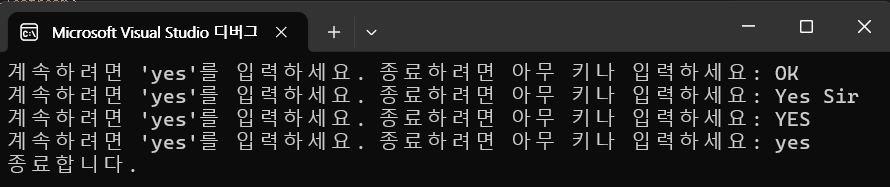
**<실행결과>**

종료하고 싶으면 yes를 입력하세요>> OK

종료하고 싶으면 yes를 입력하세요>> Yes Sir

종료하고 싶으면 yes를 입력하세요>> yes

종료합니다..

* 소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)
* 
* 출력화면 캡쳐
* 

**#2-4. 중식당의 주문 과정을 C++ 프로그램으로 작성해보자. 다음 실행 결과와 같이 메뉴와 사람 수를 입력받고 이를 출력하면 된다. (잘못된 입력을 가려내는 부분도 코드에 추가하라)**

**<실행결과>**

\*\*\*\*한국공학반점에 오신 것을 환영합니다\*\*\*\*

짬뽕:1, 짜장:2, 군만두:3, 종료:4>> 1

몇인분? 4

짬뽕 4인분 나왔습니다.

짬뽕:1, 짜장:2, 군만두:3, 종료:4>> 2

몇인분? 6

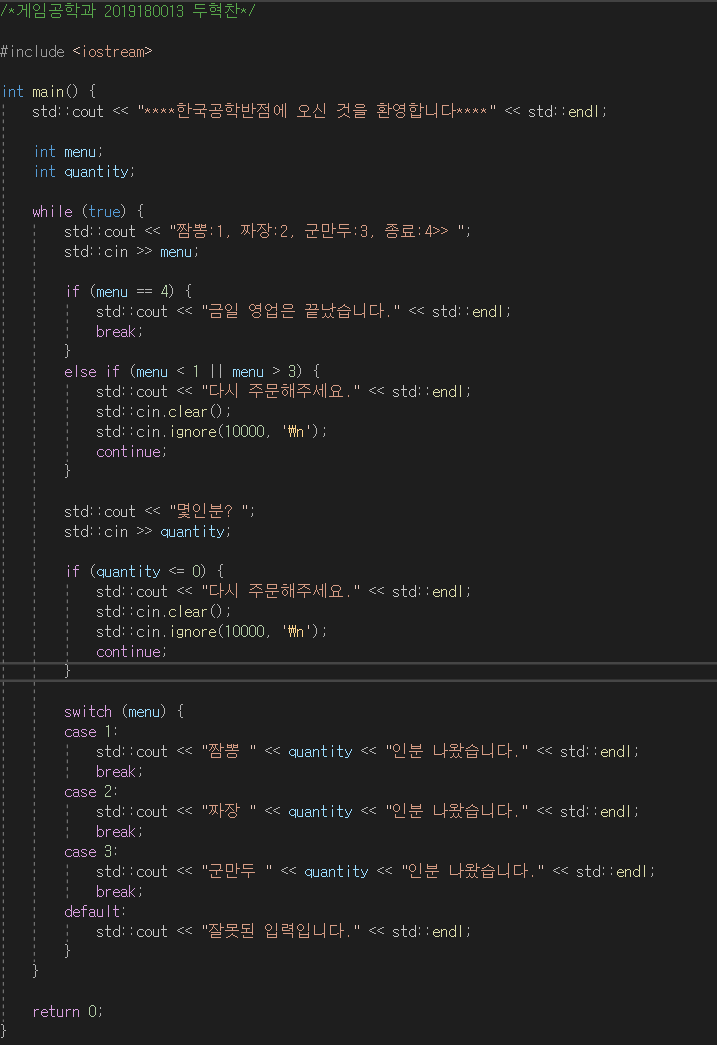
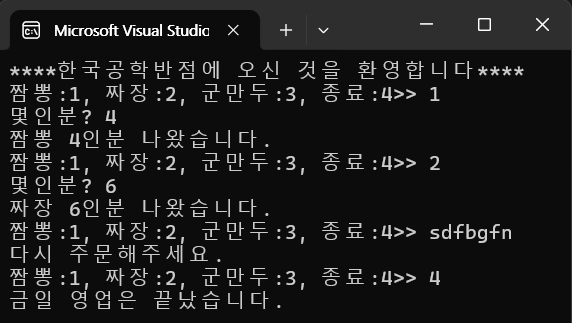
짜장 6인분 나왔습니다.

짬뽕:1, 짜장:2, 군만두:3, 종료:4>> 6

다시 주문해주세요.

짬뽕:1, 짜장:2, 군만두:3, 종료:4>> 4

금일 영업은 끝났습니다.

* 소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)
* 
* 출력화면 캡쳐
* 

**#3-1. main()의 실행 결과가 다음과 같도록 Tower 클래스를 작성하라.**

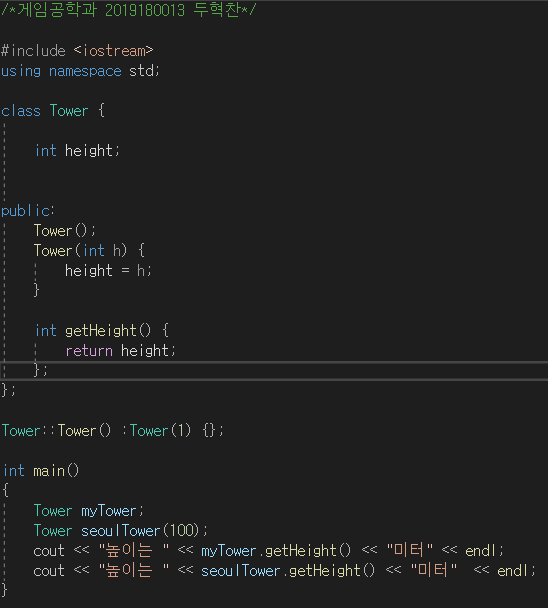
|  |
| --- |
| #include <iostream>  using namespace std;  int main()  {  Tower myTower;  Tower seoulTower(100);  cout<<”높이는”<<myTower.getHeight()<<”미터”<<endl;  cout<<”높이는”<<seoulTower.getHeight()<<”미터”<<endl;  } |

**<실행결과>**

높이는 1미터

높이는 100미터

* 소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)



* 출력화면 캡쳐



**#3-2. 은행에서 사용하는 프로그램을 작성하기 위해, 은행 계좌 하나를 표현하는 클래스 Account가 필요하다. 계좌 정보는 계좌의 주인, 계좌 번호, 잔액을 나타내는 3 멤버 변수로 이루어진다. main() 함수의 실행 결과가 다음과 같도록 Account 클래스를 작성하라.**

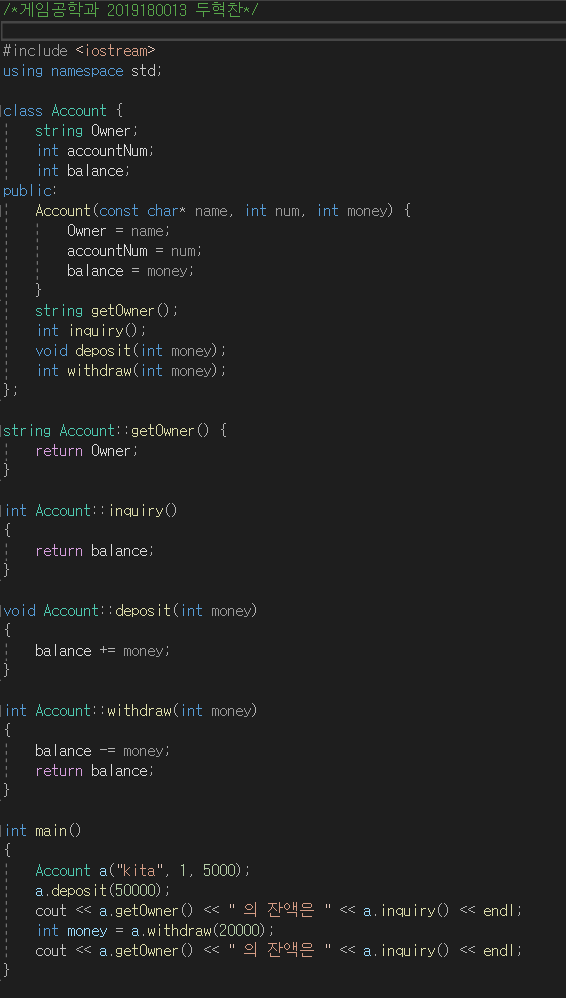
|  |
| --- |
| int main()  {  Account a(“kitae”, 1, 5000);  a.deposit(50000);  cout<<a.getOwner()<<”의 잔액은 ”<<a.inquiry()<<endl;  int money = a.withdraw(20000);  cout<<a.getOwner()<<”의 잔액은”<< a.inquiry()<<endl;  } |

**<실행결과>**

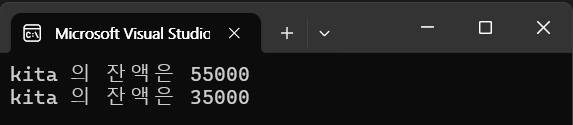
kitae의 잔액은 55000

kitae의 잔액은 35000

소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)



* 출력화면 캡쳐



**#3-3. Oval 클래스는 주어진 사각형에 내접하는 타원을 추상화한 클래스이다. Oval 클래스의 멤버는 모두 다음과 같다. Oval 클래스를 선언부와 구현부로 나누어 작성하라.**

* 정수값의 사각형 너비와 높이를 가지는 width, height 변수 멤버
* 너비와 높이 값을 매개 변수로 받는 생성자
* 너비와 높이를 1로 초기화하는 매개 변수 없는 생성자
* width와 height를 출력하는 소멸자
* 타원의 너비를 리턴하는 getWidth() 함수 멤버
* 타원의 높이를 리턴하는 getHeight() 함수 멤버
* 타원의 너비와 높이를 변경하는 set(int w, int h) 함수 멤버
* 타원의 너비와 높이를 화면에 출력하는 show() 함수 멤버

|  |
| --- |
| int main()  {  Oval a, b(3, 4);  a.set(10, 20);  a.show();  cout<<b.getWidth()<<”, ”<<b.getHeight()<<endl;  } |

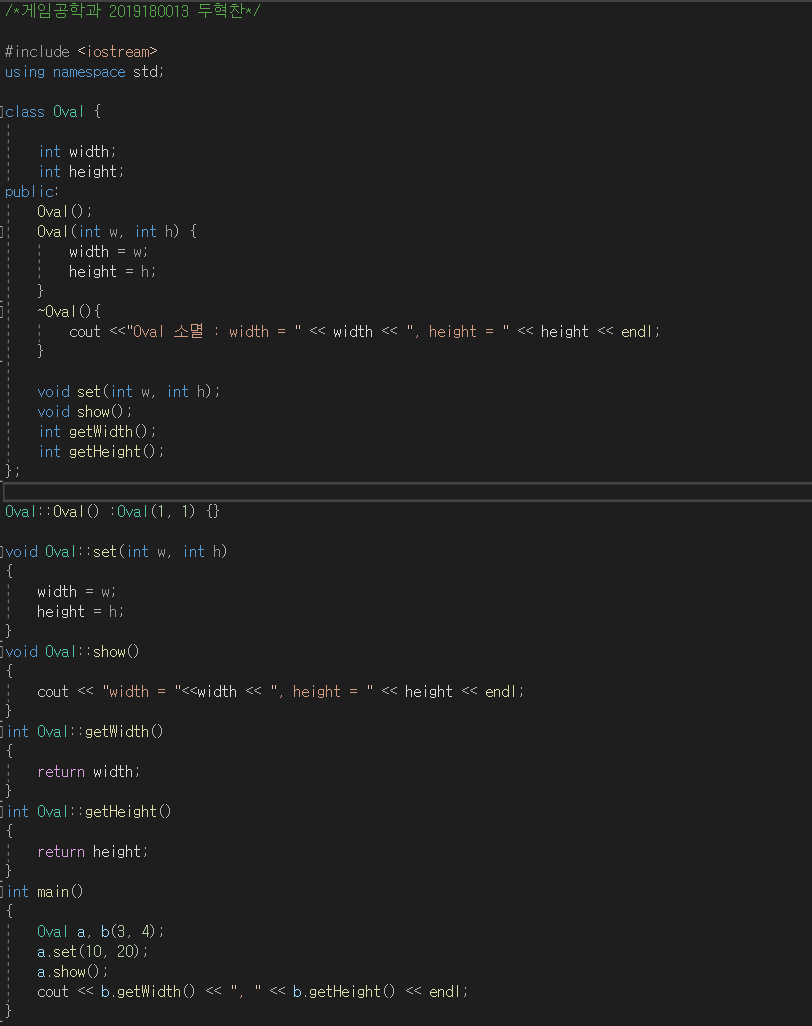
**<실행결과>**

width = 10, height = 20

3, 4

Oval 소멸 : width = 3, height = 4

Oval 소멸 : width = 10, height = 20

* 소스코드 캡쳐 (/\*스크립트 상단에 학과, 학번, 이름 입력\*/)
* 
* 출력화면 캡쳐
* 