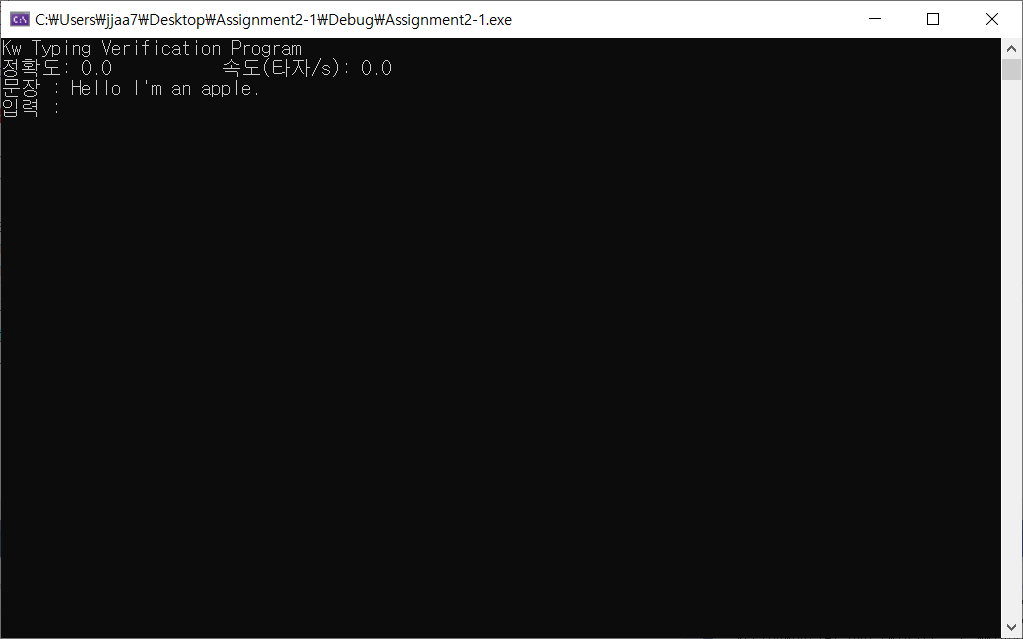
1\_A\_2019202052\_김호성\_ver2.

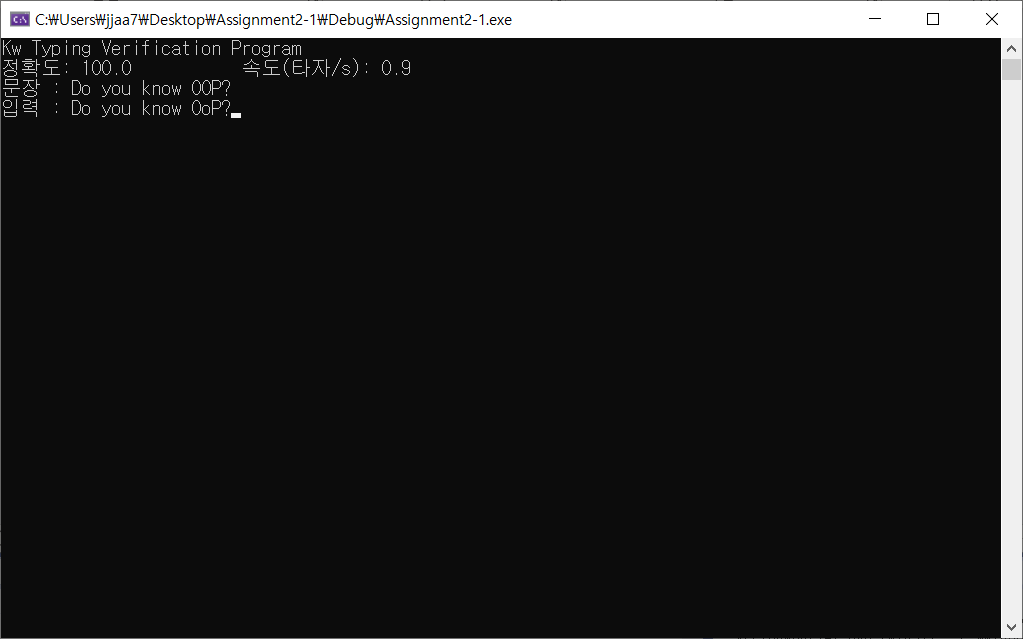
Project1

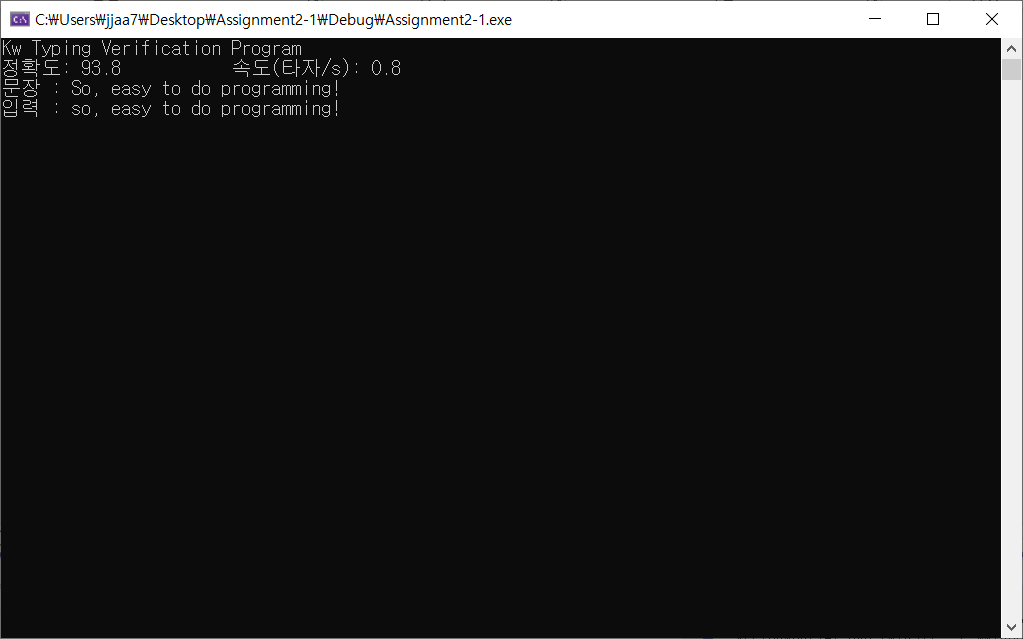
문제 및 설명

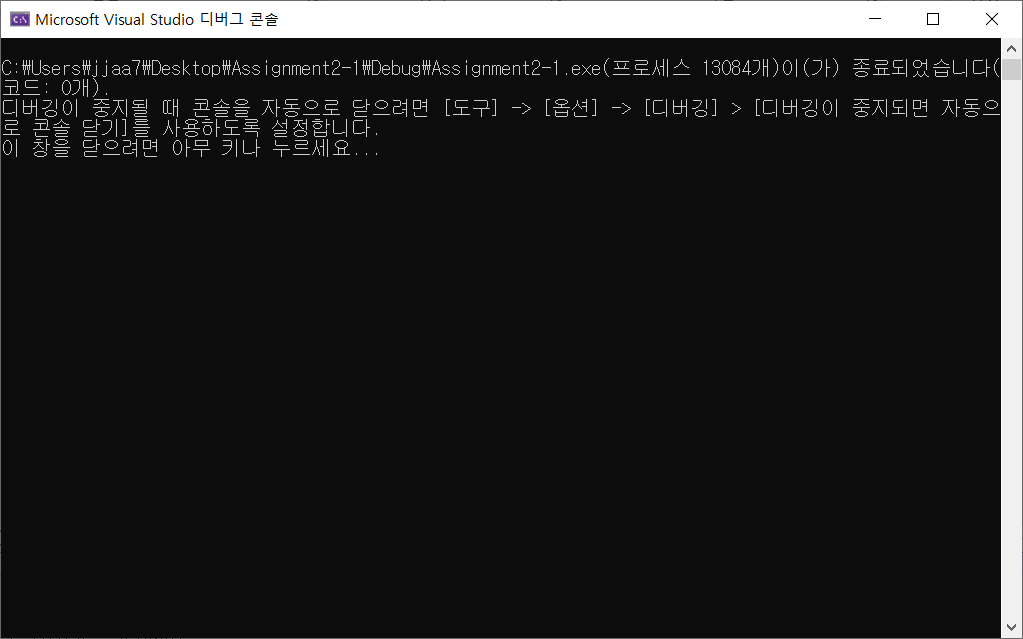
한컴 타자 연습과 유사하게 파일을 불러온 후 사용자가 입력한다. 두 문장을 비교한 후 정확도와 타자속도를 계산한다. 이후 result파일에 저장한다.

결과화면

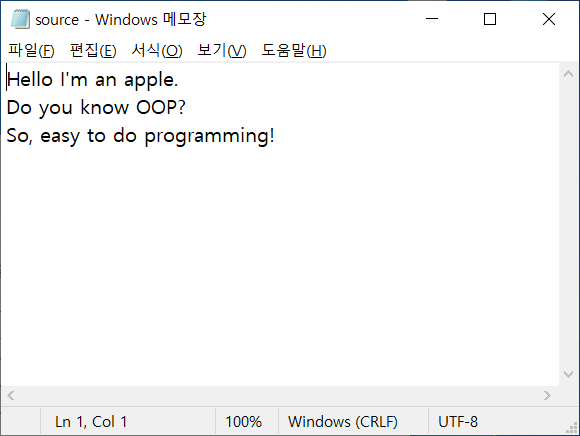








입력 파일



결과 파일



고찰

정확도 공식: count\*100 / strlen(str2)

타속 공식: c / (end - start)

소수점 한 자리 고정 방법: cout << fixed; cout.precission(1);

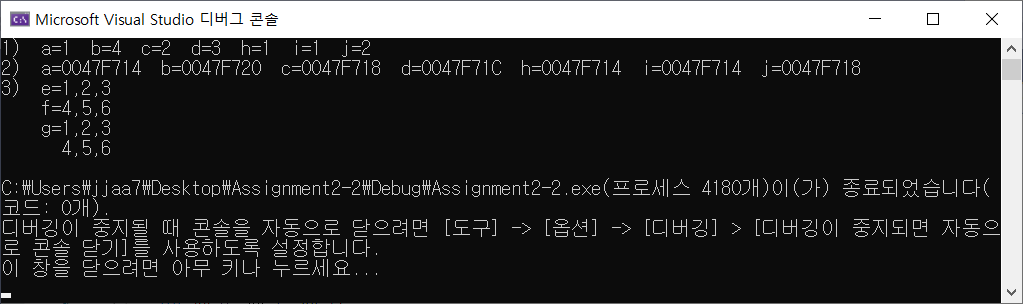
화면 전환하는 방법: system(“cls”);

Project 2

문제 및 설명

2X3배열을 생성하고 단일 포인터, 이중 포인터의 관계를 관찰한다.

결과화면



고찰

주소위치를 알고 싶다면 n중 포인터에서 포인터를 하나 지우면 된다.

주소 값을 알고 싶다면 n중 포인터 그대로 출력하면 된다.

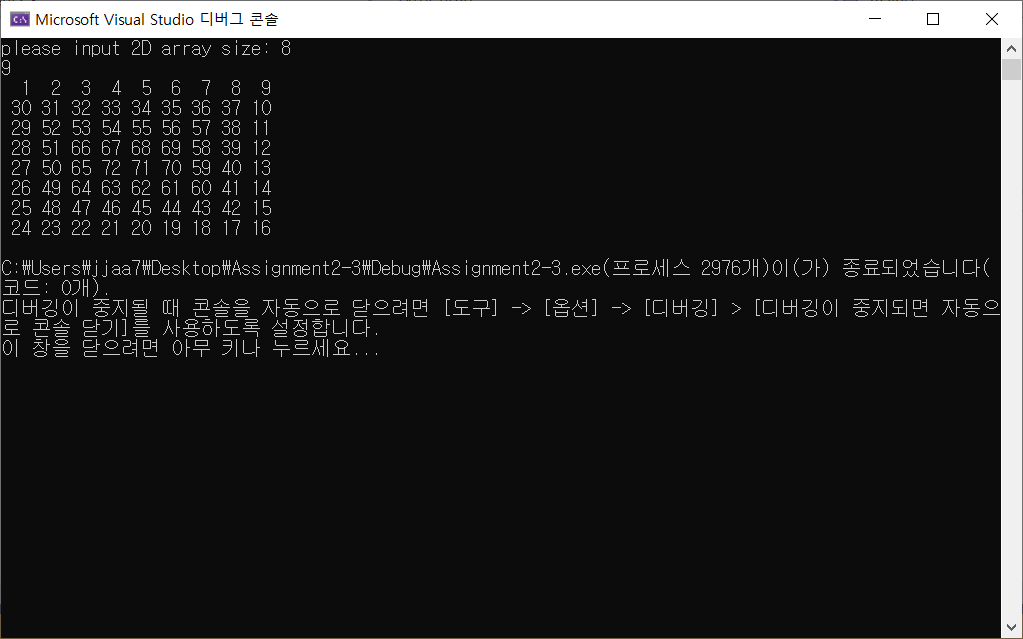
Project 3

문제 및 설명

가장자리부터 값을 채우는(중앙이 최대 값이 되는) 배열을 출력한다.

(단, 행\*열 값보다 큰 값이 올 수 없으며, 최대 크기는 10\*10이다.)

결과화면



고찰

먼저 가장자리의 윗부분을 먼저 채운다. (행1~n까지) 이후 행 -1처리를 한 후 방향 전환을 한다.

다음 반복해서 열을 채운 후 열 -1처리를 한 후 방향 전환을 한다.

반복해서 가장 안 쪽까지 채운 후 복사해둔 행, 열 값을 이용해 출력한다.

Project 4

문제 및 설명

결과화면

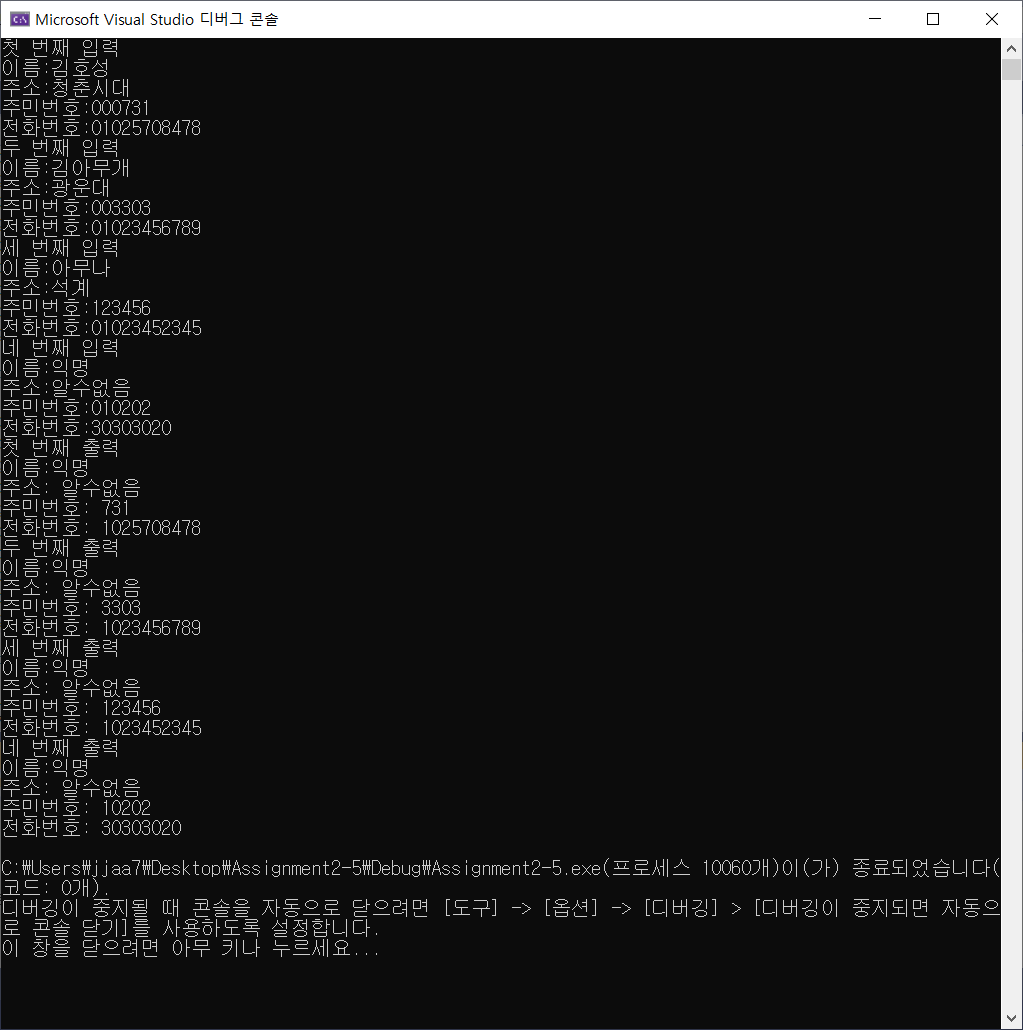
고찰

Project 5

문제 및 설명

클래스를 사용해서 4명의 개인정보를 입력 받은 후 출력한다.

결과화면



고찰

클래스의 기초를 사용한 후 객체로 만든다.

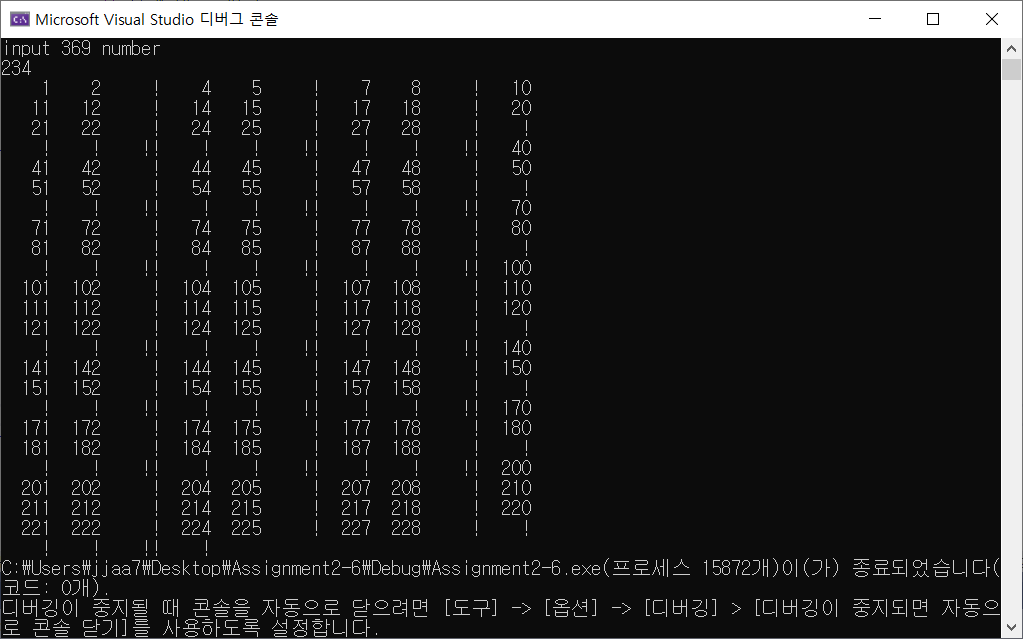
현재 코드는 불필요하게 길지만 반복문을 사용해 줄일 수 있다.

Project 6

문제 및 설명

메인 함수에서 369의 max(최대 300)값을 입력 받은 후 class내부에서 게임을 실행한다.

결과화면



고찰

Iomanip 헤더파일을 추가해 cout.setf(ios::right)를 사용한다.(오른 쪽 정렬)

숫자를 입력 받은 후 100의 자리부터 박수를 쳐야 하는지, 10의 자리부터 박수를 쳐야 하는지, 1의 자리부터 박수를 쳐야 하는지, 숫자를 말해야 하는지 나눈 후 숫자일 경우 setw(n)을 설정한다.

숫자를 입력 받은 후 자릿수에 맞게 1차원 배열에 집어넣는다.

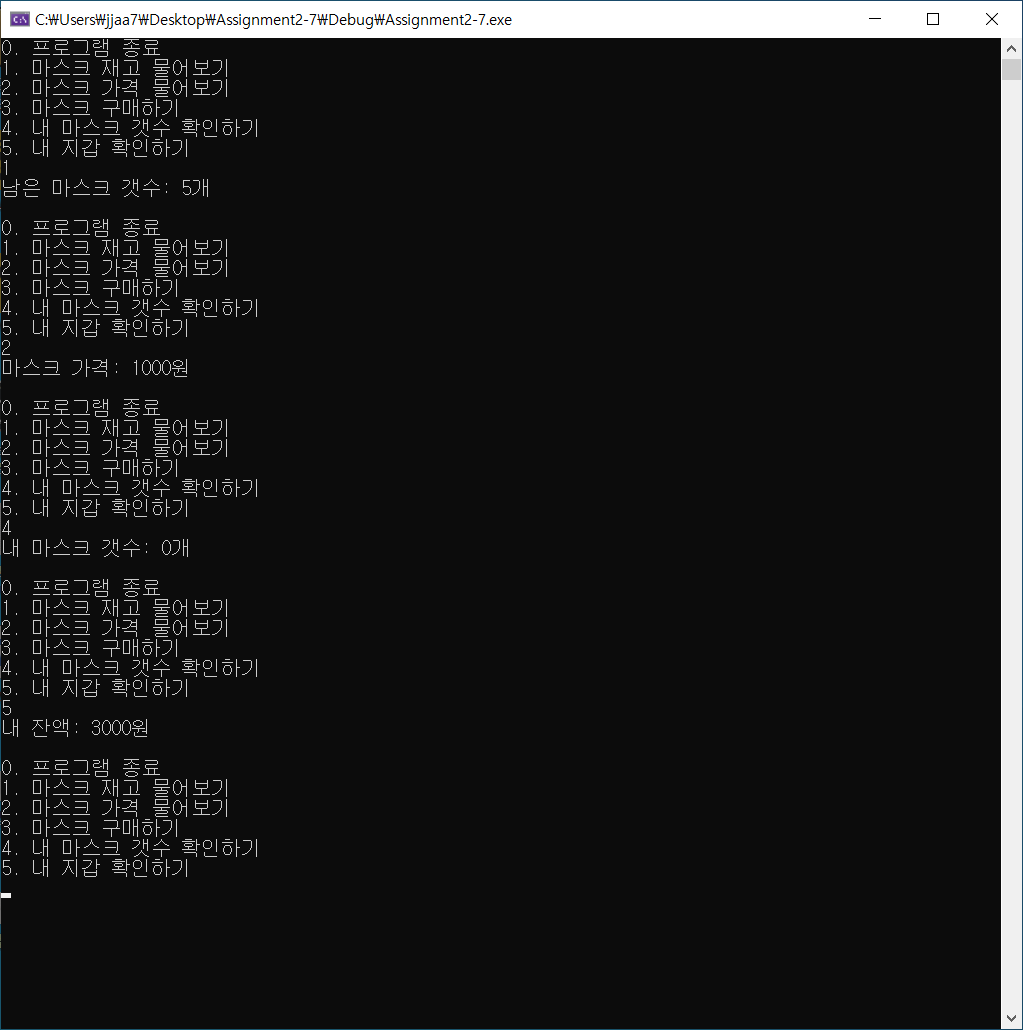
1~300사이의 입력 값이 아닌 경우 예외처리 한다.

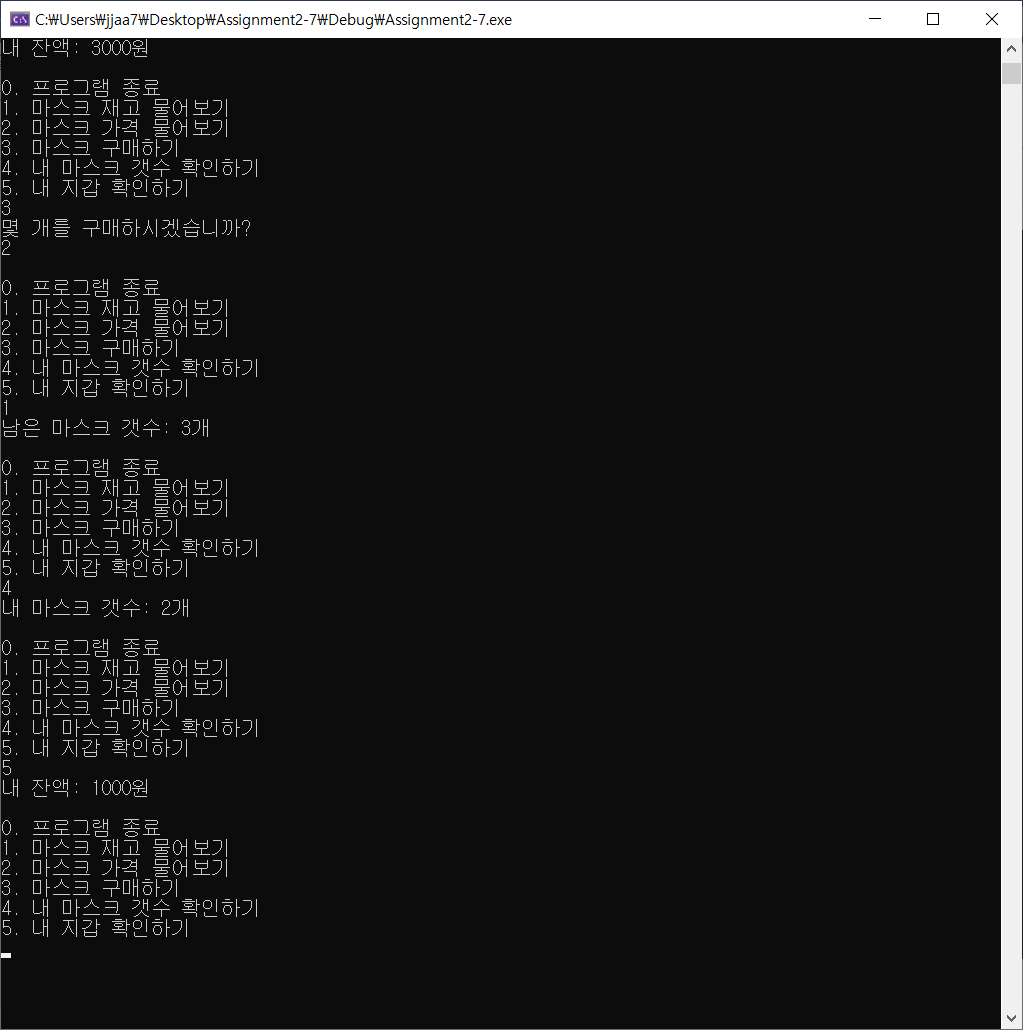
Project 7

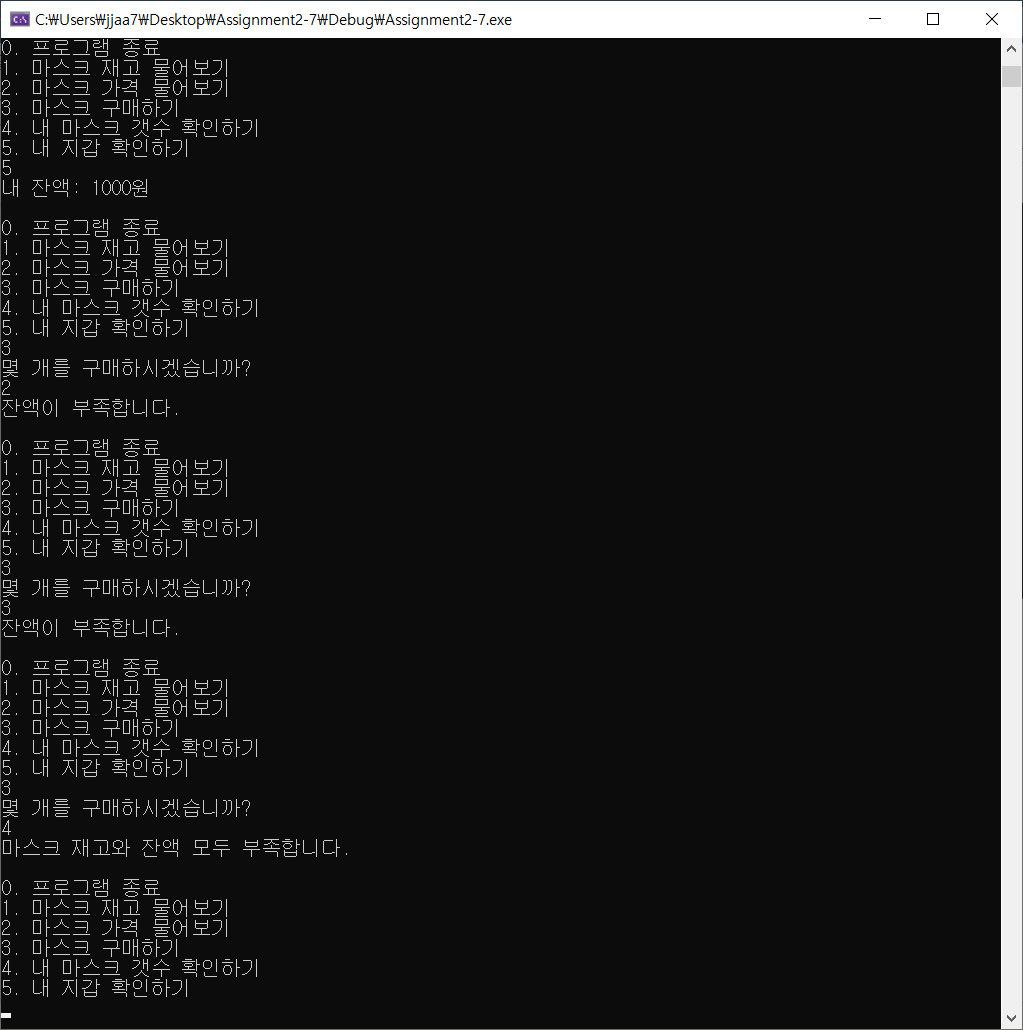
문제 및 설명

Pharamacist, Buyer class를 만들고 마스크 매매 환경을 개발한다.

결과화면







고찰

초기 값을 약사가 가지고 있는 마스크는 5개이며 가격은 1000원, 구매자가 가지고 있는 돈은 3000원 가지고 있는 마스크는 0으로 잡은 후 각자의 class에 맞게 넣어준 후 생성자를 만든다.

(이후 B\_mask로 인해 값이 자동으로 변한다.)

잔액 < 구매하려는 마스크 양 \* 금액 일 경우 잔액이 부족하다고 뜨며,

마스크 재고 < 사려는 양 일 경우는 재고가 부족하다고 뜬다.(초기 값 변경 시 가능)

둘 다 조건에 만족 하면 재고와 잔액 모두 부족하다고 뜬다.

Project 8

문제 및 설명

결과화면

고찰

Project 9

문제 및 설명

결과화면

고찰