알고리즘 주별 과제 #1

2018019116 독어독문학과

이세하

과제 내용: 3가지의 정렬 기법에 대한 알고리즘은 C언어로 푼 뒤 결과 값 추출

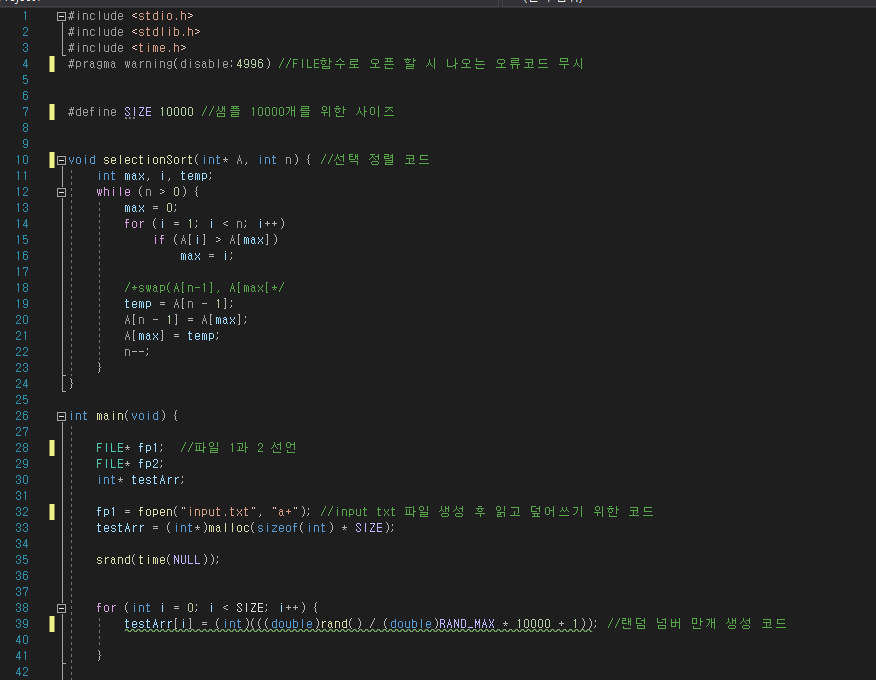
중요한 점: 파일 입, 출력에 대한 이해도가 필요하다.

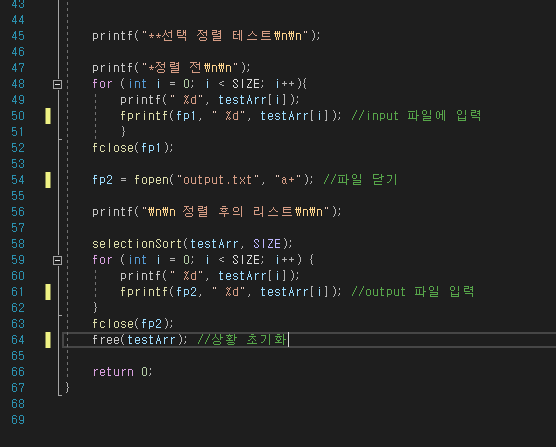
Elements: rand()를 이용해 무작위 숫자 10,000개를 추출.

실험 코드: \*세가지 기법을 한 코드에 넣다 보니 결과값을 얻는 시간이 너무 오래 걸려서 세 개의 코드로 나눠서 풀었다.

구현한 환경: Visual Studio 2019

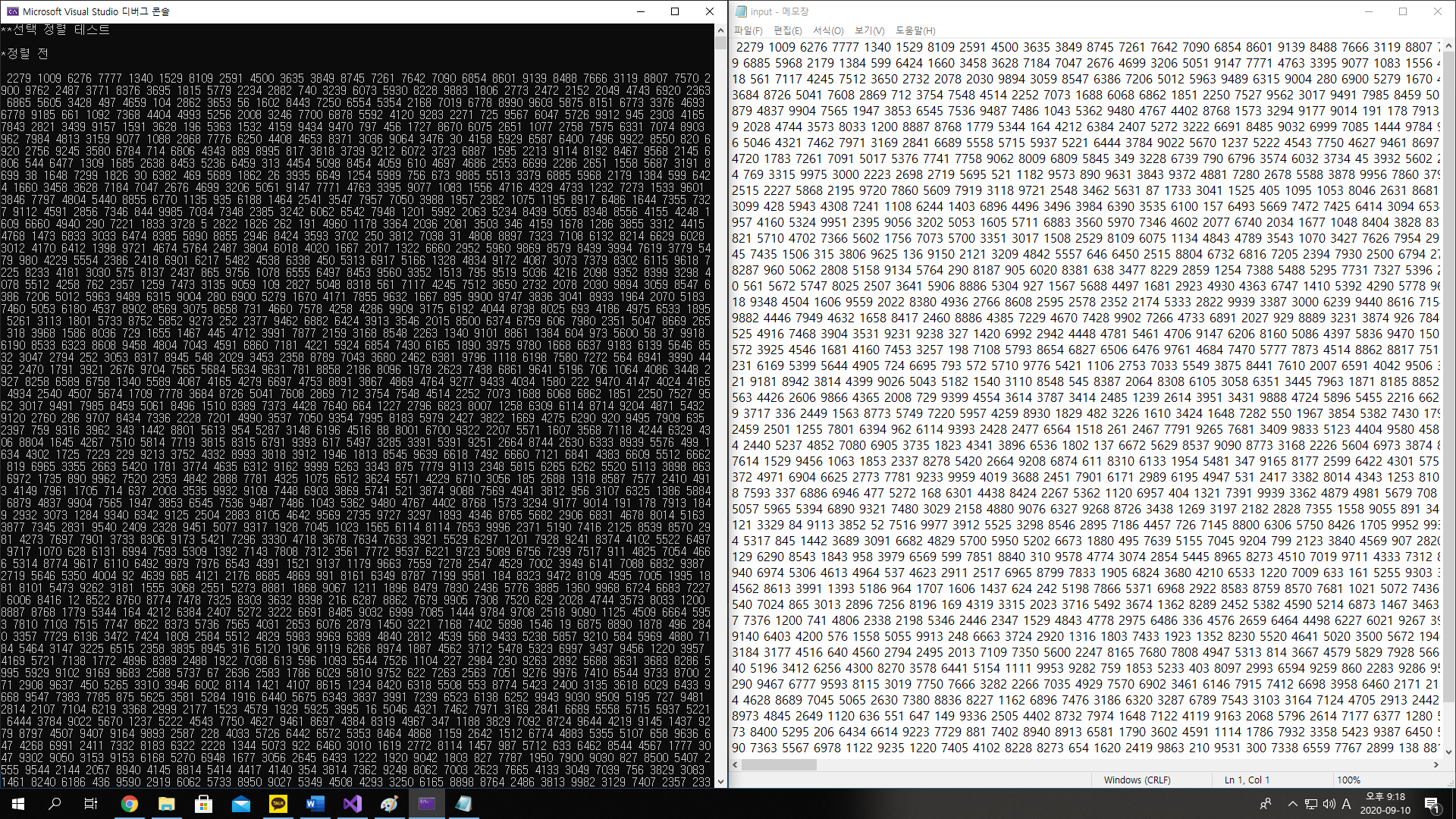
코드1: 선택정렬

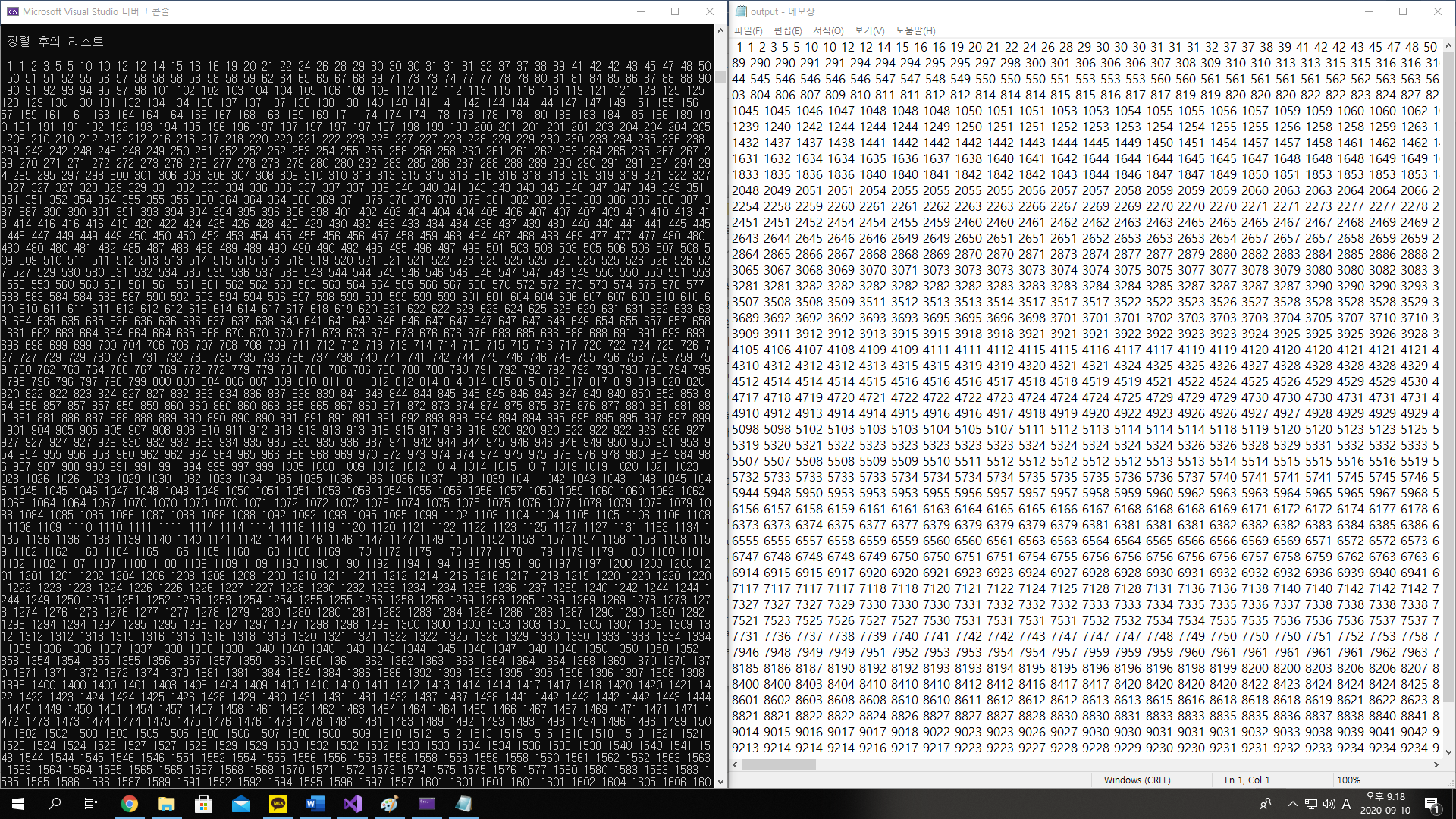




각각의 코드 설명은 옆 주석을 통해 작성했다. 그 중 선택 정렬 코드는 알고리즘 강의 1에 나온 코드를 차용했다. 그 뒤 파일 1과 2 선언 방법과 입출력, 그리고 테스트하는 코드와 정렬 후의 값을 뽑는 코드는 구글링을 통해 인용했다.

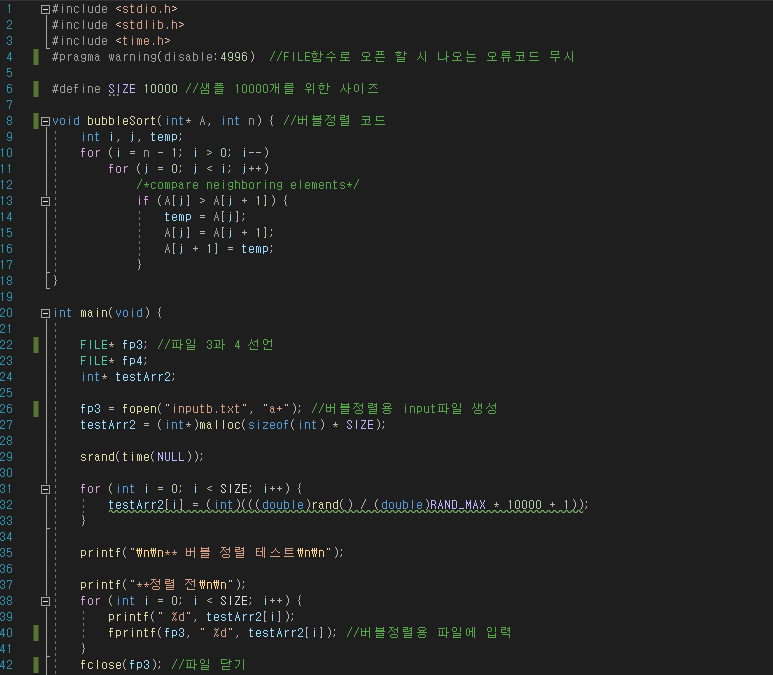
결과값:

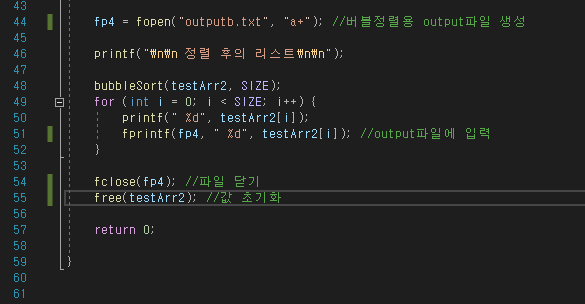




다음과 같이 cmd창에 나온 값과 txt파일이 생성 후 입력된 값이 서로 같고 큰 문제 없이 결과값이 나왔다.

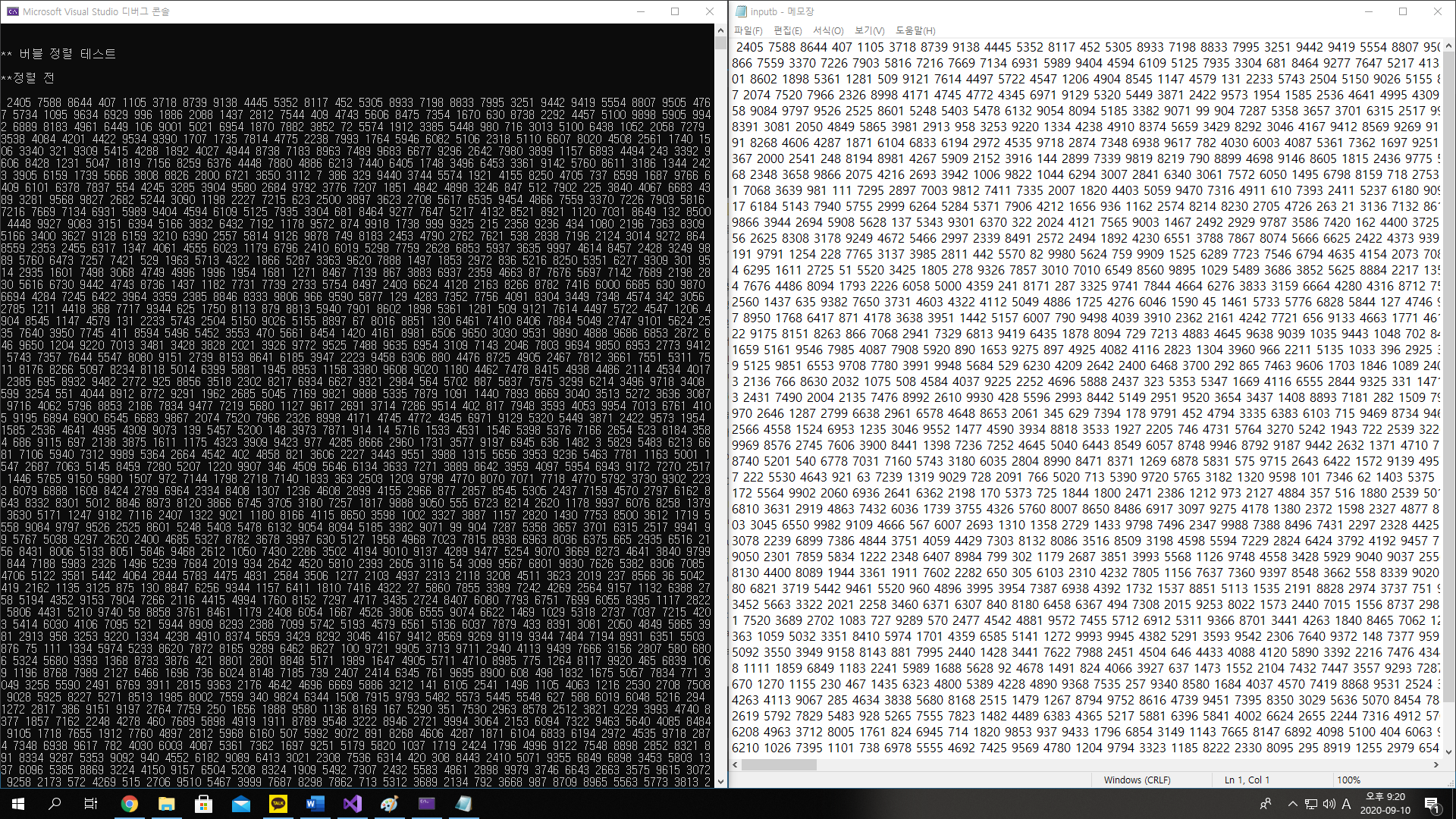
코드2: 버블정렬

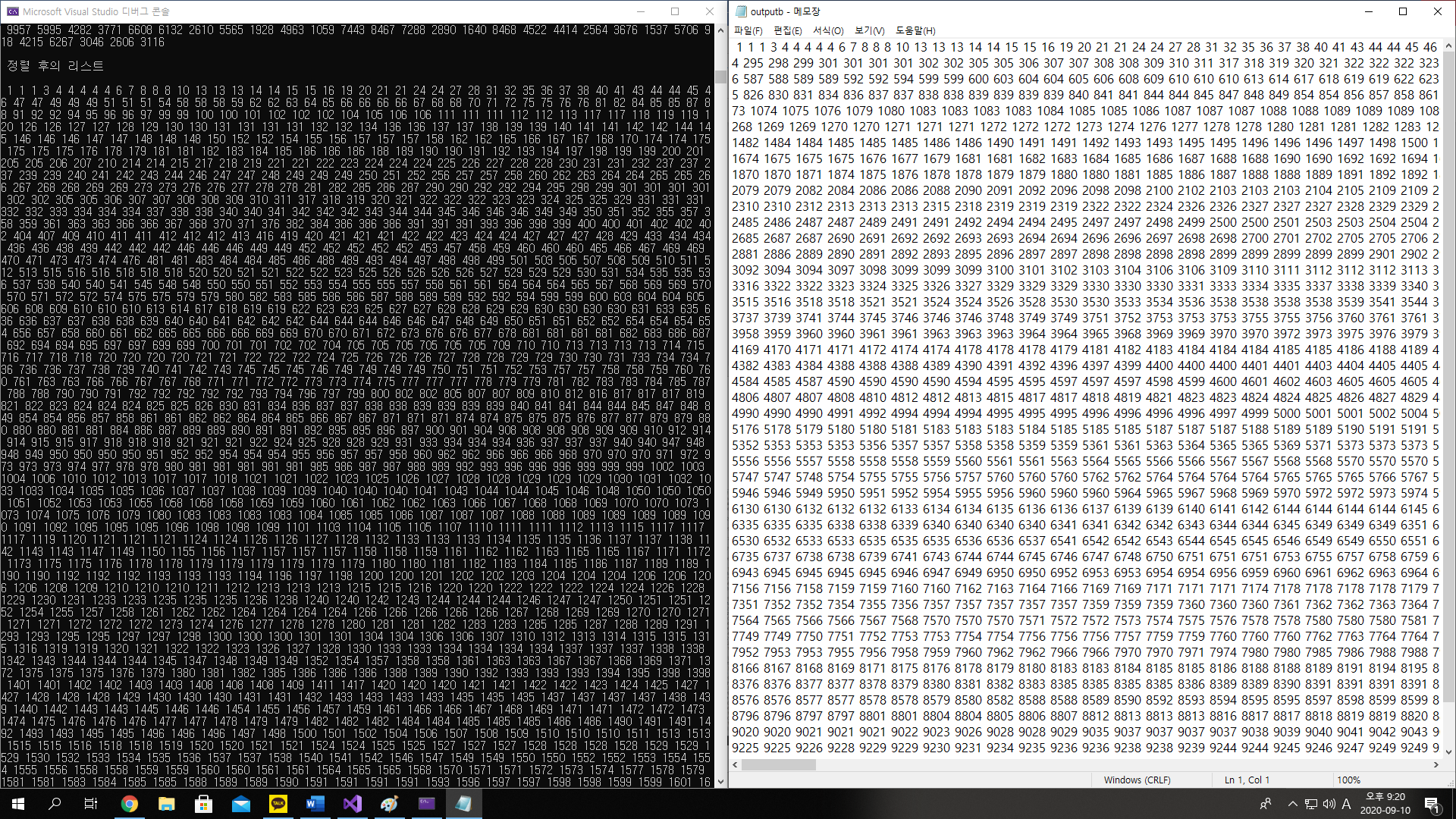




마찬가지로 int main()부분은 다른 정렬코드와 다를 바 없이 똑같이 만들고, void bubbleSort()부분만 알고리즘 수업 자료에 나온 코드를 차용하여 작성했다.

결과:

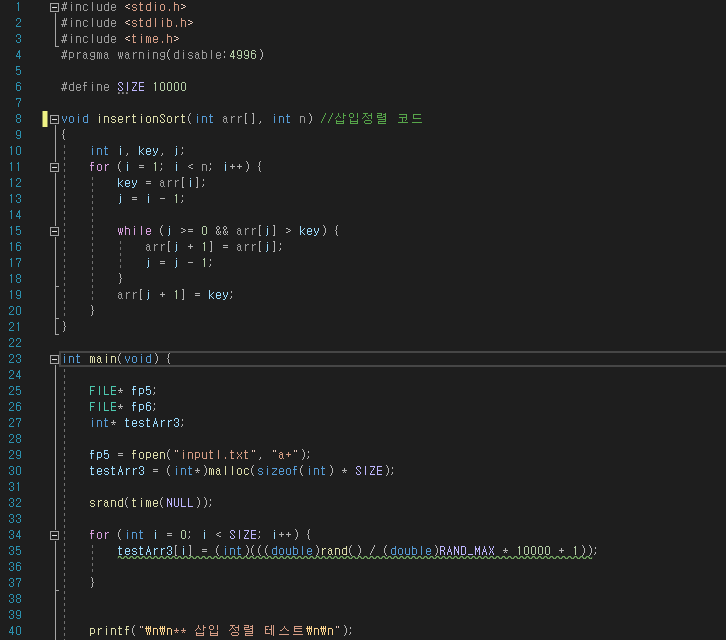




마찬가지로 결과값이 나온 cmd창과 txt 파일 값이 같고 오류 없이 정렬이 잘 됐다는 것을 확인할 수 있다.

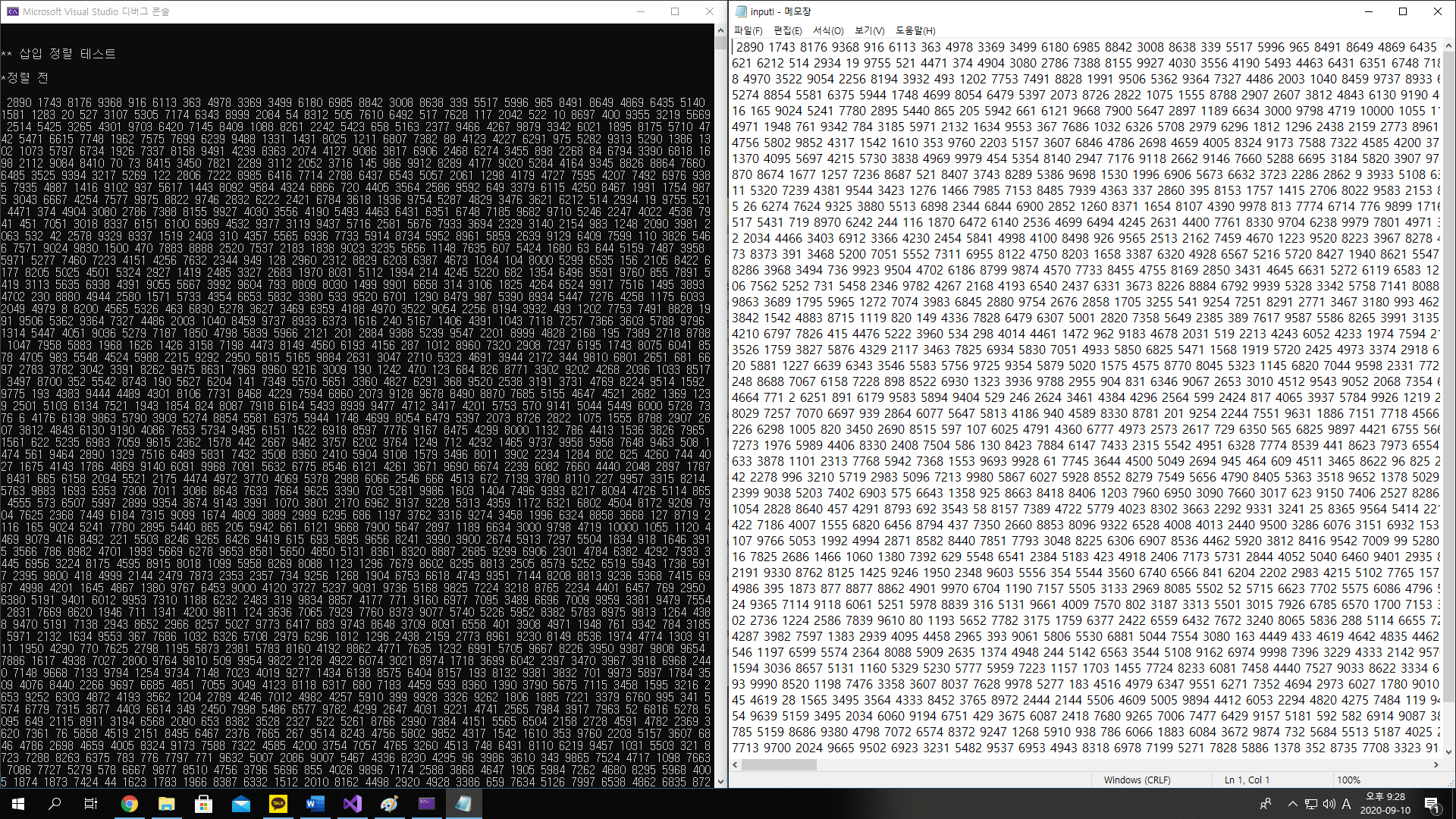
코드3: 삽입 정렬

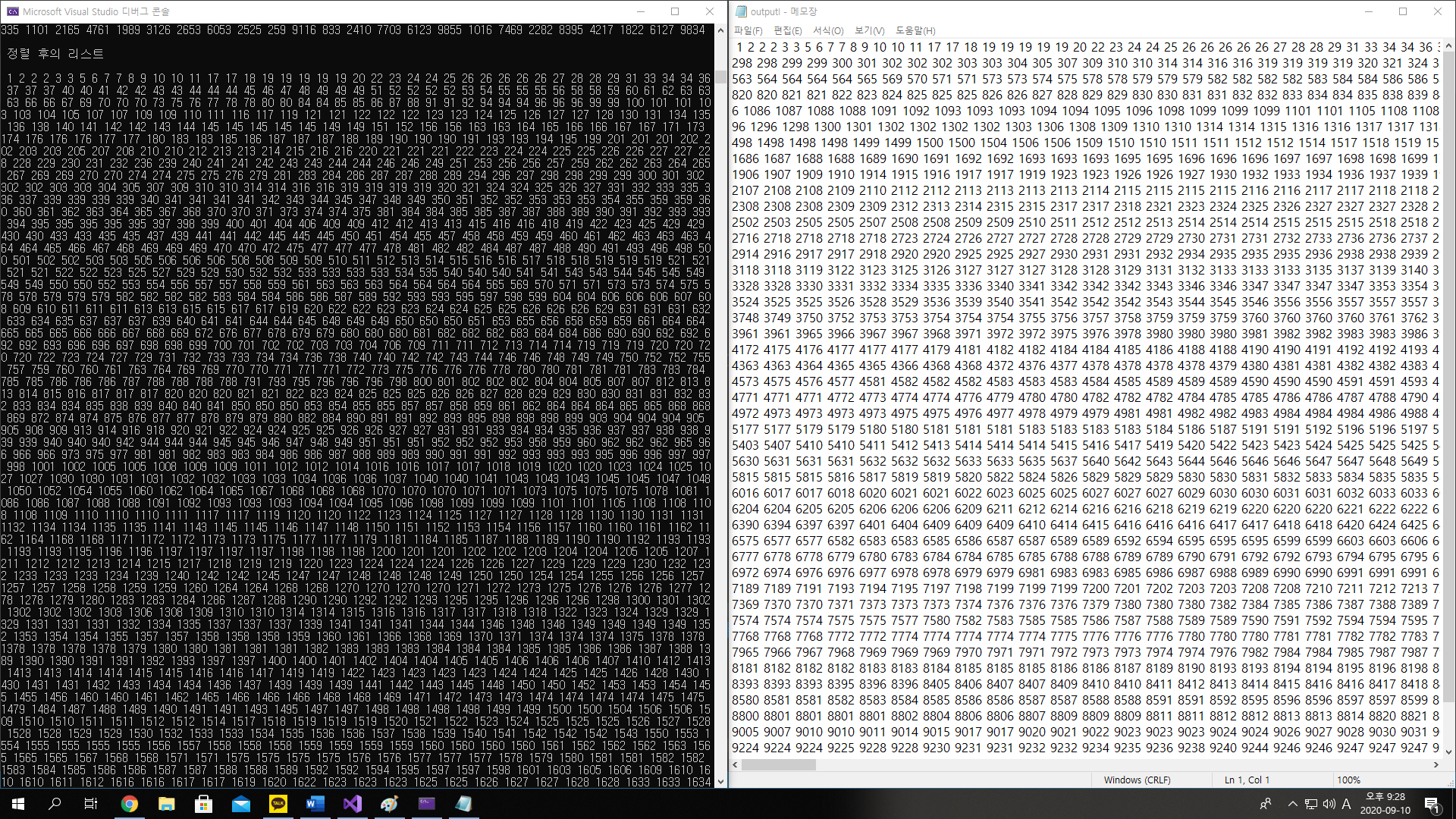
마찬가지로 같은 방법의 main()함수 코드들을 사용했지만 요번에는 삽입정렬 코드가 수업 자료안의 코드와 살짝 다르다. 처음에는 수업 자료 안 코드를 사용했지만 결과값이 계속 나오지가 않아 구글링을 통해 공부하던 중, 한 사람의 블로그에 나온 코드를 활용해 for(j)보다 while문을 사용하니 만족스러운 결과값이 나왔다.





결과:





아쉬운 점: 코딩을 하면서 좀더 깔끔하고 보기 편하게 만드는 것이 조금 부족한 것 같다. 그리고 워낙 자주 디버깅을 하다 보니 project라는 파일을 디버깅 횟수 때문에 무거운 프로그램이 된 것이 안타깝다.

참조한 사이트:

구현 코드에 대해 참조: <https://box0830.tistory.com/106>

삽입정렬 코드 문제에 대한 참조: <https://www.geeksforgeeks.org/insertion-sort/>

파일 입출력 코드 만드는 방법과 예시 참조: <https://cboard.cprogramming.com/c-programming/149997-creating-text-file-random-numbers.html>

<https://dojang.io/mod/page/view.php?id=610>