**(30점) 5.**

(10점) 5-A: ((800, 'Apple'), (50, 'Chestnut'), (1000, 'Grape'), (300, 'Persimmon'), (1500, 'Pear'))

(10점) 5-B: [['Apple', 800], ['Chestnut', 50], ['Grape', 1000], ['Persimmon', 300], ['Pear', 1500]]

(10점)5-C: {800: 'Apple', 1000: 'Grape', 50: 'Chestnut', 300: 'Persimmon', 1500: 'Pear'}

**5-A,B,C 해답에서 item의 순서는 바뀌어도 괜찮음**

**(30점) 6.**

(7점) 6-A:

90 Korean Math. History

85 English Science

95 Economics

(7점) 6-B: 90 ['Korean', 'Math.', 'History']

(7점) 6-C: ['English', 'Science']

(7점) 6-D: ['Economics']

**(30점) 7.**

(5점) 7-A: Math. 75

(5점) 7-B: english ? Not in the dictionary!

(8점) 7-C: 문자열 “quit”를 입력한다.

(12점) 7-D: 각 행에 과목명과 점수가 기록된 text file을 읽어서 과목 key로 하고 점수를 value로 하는 dictionary 만들고 과목 이름이 입력으로 주어지면 그 과목에 해당하는 점수를 화면에 출력한다. 과목이 없으면 없다는 message를 출력한다. 문자열 “quit”를 입력하면 정지한다.

**(40점) 8.**

(16점) 8-A:

Carrol Johnson 80 98

Joseph Shin 75 90

Henry Aaron 90 85

James Aaron 85 85

Goerge Hahn 80 70

David Lee 70 70

(8점) 8-B: 각 행에 성명, 국어 점수, 수학 점수가 기록 된 Text file, "mid\_term2.txt" 읽어서

점수들을 숫자로 바꾼 후 이 데이터부터 이름, 수학 점수, 과학 점수로 구성된

tuple을 만들고 이를 요소로 하는 list를 구축하여 돌려 준다.

(8점) 8-C: Tuple을 원소로 하는 list가 주어질 때 새로운 tuple, (“Carrol Johnson”, 80, 98)을

추가하고 이 list의 원소들을 과학 점수 별, 수학 점수 별 크기의 역순으로

정렬하여 돌려 준다.

(8점) 8-D: “mid\_term2.txt” text file을 읽어서 이름, 수학 점수, 과학 점수로 구성된 tuple을

원소로 하는 list를 만들고 새로운 tuple (“Carrol Johnson”, 80, 98)을 추가한 후

이 list를 과학 점수 별, 수학 점수 별 역순으로 정렬하여 화면에 출력한다.