

Computing Foundations for Data Science

HW 3

제출기한: 2021/09/23 10:59AM

주의사항

- 코드를 Jupyter Notebook 에서 작성하였더라도 python 파일(.py)로 제출할 것.
- 함수가 의도한 값을 return 하는지를 확인할 것. (print 와 혼동하지 말 것)
- 파일명은 P1.py ~ P10.py 를 유지하고, 해당 파일들을 HW3_학번_이름.zip 으로 압축하여 제출할 것. 예를 들면 학번이 2020-12345 이고, 이름이 Keondo Park 이라면 **HW3_2020_12345_KeondoPark.zip** 으로 압축하여 제출.
- 예시로 제시한 입력 값 외에도 조교가 임의로 생성한 입력 값으로도 코드가 잘 실행되는지 테스트할 예정.
- 뼈대 코드의 함수 이름 및 매개변수(parameter)는 변경하지 말 것.

문제 1.

정수(int)로만 구성된 리스트를 입력으로 받고, 리스트의 메서드(method)를 활용하여 아래의 지시를 순서대로 따르는 함수 P1을 완성하세요.

1. 입력으로 받은 리스트를 오름차순으로 정렬
2. 리스트에서 1234 제거
3. [4321, 2222]를 추가하여 리스트를 확장(extend)
4. 1번째 index 위치에 1111 추가 (주의: python의 index 규칙을 잘 확인할 것)
5. 리스트를 리턴

<가정>

1. 리스트는 정수로만 구성됨
2. 각 숫자들은 리스트 내에 한 번씩만 등장하여 숫자가 중복되는 경우는 없음
3. 입력되는 리스트에는 1234가 항상 존재

예시1)

```
>>> P1([1234, 1011, 3124])  
[1011, 1111, 3124, 4321, 2222]
```

예시2)

```
>>> P1([7777, 5555, 8888, 1234, 9876])  
[5555, 1111, 7777, 8888, 9876, 4321, 2222]
```

문제 2.

원소가 2개 이상인 리스트를 입력으로 받고, 리스트의 첫 번째 값과 마지막 값이 같은 경우에는 True, 다를 경우에는 False를 리턴하는 함수를 작성하세요.

<조건>

1. 입력되는 리스트의 원소는 항상 2개 이상
2. 리턴 값은 bool 타입

예시1)

```
>>> P2([2, 9, 4, 8, 2])  
True
```

예시2)

```
>>> P2(['computer', 'keyboard', 'mouse'])  
False
```

예시3)

```
>>> P2([1.0, 1.5])  
False
```

문제 3.

세 개의 리스트를 입력으로 받고, 가장 길이가 긴 리스트의 길이를 리턴하는 함수 P3를 작성하세요.

<조건>

1. 항상 3개의 리스트가 입력으로 주어진다고 가정

예시1)

```
>>> P3([1], [2, 3], [4, 5, 6])
```

```
3
```

예시2)

```
>>> P3([10, 11, 12, 13], ['a'], [1.11])
```

```
4
```

문제 4.

정수로 구성된 리스트를 입력으로 받고 리스트 안의 값이 음수이면 양수로, 양수이면 음수로 변경하여 리턴하는 함수 P4를 작성하세요.

<조건>

1. 값이 0일 때는 변경 없음
2. 입력 받은 원본 리스트를 변경하지 말고, 새로운 리스트를 리턴하여야 함

예시1)

```
>>> P4([-1, 2])
```

```
[1, -2]
```

예시2)

```
>>> P4([66, 0, -11])
```

```
[-66, 0, 11]
```

문제 5.

중첩 리스트(nested list)를 입력으로 받고, 아래 조건을 만족하는 리스트를 리턴하는 함수 P5를 작성하세요.

<조건>

1. 입력 리스트는 [[영단어, 글자수]] 꼴이며, 자세한 건 아래의 예시를 참고
2. 영단어는 알파벳 소문자로만 이루어져 있음
3. 입력 리스트에서 단어만 모아서 리스트로 리턴
4. 리스트는 알파벳 순으로 정렬되어 있어야함

예시1)

```
>>> P5(['apple', 5], ['banana', 6])  
['apple', 'banana']
```

예시2)

```
>>> P5(['cup', 3], ['ace', 3], ['nice', 4], ['good', 4])  
['ace', 'cup', 'good', 'nice']
```

문제 6.

알파벳으로만 구성된 리스트를 입력으로 받고, 대문자가 처음으로 등장하는 index를 리턴하는 함수 P6을 작성하세요. (for loop을 통해서 해결 가능하지만, while loop을 활용하는 훈련을 해보는 것도 좋습니다. 어떻게 구현하여도 점수와 관계는 없습니다.)

<조건>

1. 입력 리스트는 알파벳 대문자 혹은 소문자로만 구성됨
2. 대문자가 존재하지 않는 경우는 -1을 리턴

예시1)

```
>>> P6(['a', 'B', 'c', 'D'])  
1
```

예시2)

```
>>> P6(['z', 'b', 'k', 'D', 'l', 'q'])  
3
```

예시3)

```
>>> P6(['a', 'b'])  
-1
```

문제 7.

정수로만 구성된 리스트를 입력으로 받고, 리스트의 원소 중 양수만 더해서 그 합을 리턴하는 함수 P7을 작성하세요.

<조건>

1. 입력 리스트는 정수로만 구성됨
2. 리스트에 양수가 존재하지 않는 경우는 0을 리턴

예시1)

```
>>> P7([1, 2, -3])
```

```
3
```

예시2)

```
>>> P7([-1, 0])
```

```
0
```

문제 8.

정수로만 구성된 길이가 같은 리스트 2개를 입력으로 받고, 리스트의 각 인덱스 별로 값을 비교하여 큰 값을 해당 인덱스에 대한 값으로 갖는 리스트를 리턴하는 함수 P8을 작성하세요.

<조건>

1. 입력 리스트 2개는 각각 정수로만 구성되고 길이가 서로 같음
2. 비교 결과 값이 같을 경우는 둘 중 아무 값이나 활용

예시1)

```
>>> P8([1, 3], [2, 0])
```

```
[2, 3]
```

예시2)

```
>>> P8([1, 2, 3], [0, 5, 10])
```

[1, 5, 10]

예시3)

>>> P8([1, 3], [1, 2])

[1, 3]

문제 9.

정수 2개를 입력으로 받고, 아래 조건을 만족하는 리스트를 리턴하는 함수 P9를 작성하세요.

<조건>

1. 입력은 항상 정수 2개
2. 입력 받은 두 정수 중 큰 수부터 시작하여 작은 수까지 포함된 리스트를 리턴
3. 리턴하는 리스트는 내림차순으로 정렬되어 있어야 함
4. 입력 받는 두 정수가 같은 경우는 해당 정수가 1번만 포함된 리스트를 리턴

예시1)

>>> P9(3, 9)

[9, 8, 7, 6, 5, 4, 3]

예시2)

>>> P9(2, -1)

[2, 1, 0, -1]

예시3)

>>> P9(100, 100)

[100]

문제 10.

영단어 리스트를 입력으로 받고, 아래 조건을 만족하는 리스트를 리턴하는 함수 P10를 작성하세요.

<조건>

1. 입력 리스트는 영단어로만 구성
2. 영단어 중에서 'a' 혹은 'A'로 시작하는 단어를 입력 순서대로 리턴
3. 'a' 혹은 'A'로 시작하는 단어가 없는 경우는 빈 리스트를 리턴

예시1)

```
>>> P10(['cat', 'ANN', 'dice', 'any', 'tire'])
```

```
['ANN', 'any']
```

예시2)

```
>>> P10(['dog', 'Money'])
```

```
[]
```