

# [파이썬 트랙] 1회차 월말평가 - Python



## 시험 과목

- ✓ Python

## 시험 목적

- ✓ Python 프로그래밍 기본 문법에 대한 이해

## 시험 유의 사항

- 1) 성실하게 테스트에 임할 것 (부정 행위시 강력 조치 및 근거가 남음)
- 2) 각 문제별로 스켈레톤 코드가 작성된 파일이 제공되며, 해당 코드 파일을 수정하여 정답 코드를 작성할 것
- 3) 소스코드 유사도 판단 프로그램 기준 부정행위로 판단될 시,  
0점 처리 및 학사 기준에 의거 조치 실시 예정
- 4) 테스트 케이스와는 별도로 채점 케이스가 존재함
- 5) 일부 문제에서 python 내장 함수(min, max, len, sum) 사용 불가  
(문제에 명시되어 있으며 사용시 해당 문제 감점 처리)
- 6) 사용자의 입력을 받는 input 함수는 절대 사용 금지
- 6) 잘못된 정답을 반환할 시 오답으로 간주하므로 반드시 문제를 잘 읽어보고 풀 것

## 시험 코드 작성 유의 사항

최종 제출 코드가 다음 항목에 해당하는 경우, 감점 혹은 0점 처리 될 수 있음

- 1) Syntax Error로 인한 채점이 불가능한 경우
- 2) input 함수를 사용하는 경우
- 3) 주석 설명이 없거나 미흡한 경우
- 4) 출력 결과에 정답과 무관하거나 불필요한 내용이 있는 경우
- 5) 문제를 풀지 못하였다면 작성한 코드를 주석 처리하거나 삭제 후 pass 키워드 작성

# [파이썬 트랙] 1회차 월말평가 - Python



## 시험 환경

- Visual studio code(이하 vscode)를 이용한다.
- 그 외 (jupyter notebook, Pycharm, ...) 사용 불가

## 코드 실행

- vscode의 터미널창에서 python 실행 명령어로 결과 확인을 추천

```
Edwin@LECTURE MINGW64 /d/SSAFY  
$ python problem01.py
```

## 정답 제출 안내

**제출 안내 사항 미 준수 시, 감점 혹은 0점 처리 될 수 있음**

1) 압축 및 제출 파일 이름

- 지역\_0반\_홍길동

ex) 서울\_1반\_홍길동 / 부울경\_2반\_김싸피

2) 압축 폴더 구조

- 시험을 진행했던 폴더 구조 그대로 압축하여 제출 진행
- 우측 구조에서 '서울\_1반\_김싸피' 폴더 선택 후 압축

```
서울_1반_김싸피/  
    problem01.py  
    problem02.py  
    problem03.py  
    ...  
    problem11.py
```

제출 마감시간에 서버 요청이 집중될 수 있으므로, 미리 제출하는 것을 권장함  
**(마감 시간 이후 제출 불가)**

# [파이썬 트랙] 1회차 월말평가 - Python



## 문제 01 (problem01.py) – 15점

---

- ❖ 김싸피는 노력 끝에 희망하던 기업으로 취직에 성공하였다.
- ❖ 회사 근처로 이사하기 위해 발품을 팔아 해당 지역의 월세를 리스트로 정리 해두었다.  
(월세는 0 이상의 자연수이다.)
- ❖ 월세 리스트를 인자로 받아 가장 낮은 월세를 반환하는 **min\_cost** 함수를 완성하시오.
- ❖ **Python 내장함수 min 사용 시 감점**

## 문제 02 (problem02.py) – 15점

---

- ❖ 김싸피는 노력 끝에 희망하던 기업으로 취직에 성공하였다.
- ❖ 회사 근처로 이사하기 위해 발품을 팔아 해당 지역의 월세를 리스트로 정리 해두었다.  
(월세는 0 이상의 자연수이다.)
- ❖ 월세 리스트를 인자로 받아 전체 월세의 평균을 계산하여 반환하는 **average\_cost** 함수를 완성하시오.
- ❖ **Python 내장함수 len, sum 사용 시 감점**

# [파이썬 트랙] 1회차 월말평가 - Python



## 문제 03 (problem03.py) – 10점

---

- ❖ 이싸피는 새롭게 서비스되는 사이트에서 로그인 기능을 담당하게 되었다.
- ❖ 사용자의 입력 정보는 dictionary 형태로 함수의 매개변수에 전달된다고 한다.
- ❖ 사용자의 user\_id 가 4글자 이상 16자 미만인지 확인하여 정상 범위라면 True, 그렇지 않다면 False를 반환하는 **check\_user\_id** 함수를 완성하시오.
- ❖ **Python 내장함수 len 사용 시 감점**

## 문제 04 (problem04.py) – 10점

---

- ❖ 이싸피는 새롭게 서비스되는 사이트에서 로그인 기능을 담당하게 되었다.
- ❖ 사용자의 입력 정보는 dictionary 형태로 함수의 매개변수에 전달된다고 한다.
- ❖ 사용자가 입력한 password 는 8자리 이상 32자리 미만이고, password 와 password\_confirm 도 완벽히 일치해야 한다.
- ❖ 이 두 조건을 모두 만족하면 True, 하나라도 만족하지 않으면 False를 반환하는 **compare\_pw** 함수를 완성하시오.
- ❖ **Python 내장함수 len 사용 시 감점**

# [파이썬 트랙] 1회차 월말평가 - Python



## 문제 05 (problem05.py) – 10점

- ❖ 이싸피는 새롭게 서비스되는 사이트에서 로그인 기능을 담당하게 되었다.
- ❖ 사용자의 입력 정보는 dictionary 형태로 함수의 매개변수에 전달된다고 한다.
- ❖ 사용자가 입력한 email 에 @가 포함되어 있다면 True, 포함되어 있지 않다면 False를 반환하는 **check\_email** 함수를 완성하시오.

## 문제 06 (problem06.py) – 10점

- ❖ 다음 지문은 쇼핑몰에서 판매하는 모든 아이템의 이름과 가격이 담겨진 딕셔너리이다.
- ❖ 쇼핑 카트 정보(cart\_list)가 리스트 형태로 함수의 매개변수에 전달될 때, 해당 딕셔너리를 참고하여 cart\_list 에 저장된 아이템의 총 금액을 반환하는 **calc\_total** 함수를 완성하시오.

```
item_dict= {
    'apple_pie': 2500,      'banana_milk': 1800,      'coconut_milk': 2000,
    'egg_tart': 2300,      'fruits_cocktail': 3000,      'gum': 1200,
    'hotdog': 2500,        'ice_cream': 3200,        'juice': 2800,
    'keyboard': 35000,     'lotion': 8700
}
```

## [파이썬 트랙] 1회차 월말평가 - Python



## 문제 07 (problem07.py) – 10점

- ❖ 싸피 1반 학우들이 사용하는 메일 서비스 업체들이 아래와 같이 리스트로 정리되어 있다.
- ❖ 박싸피는 각 업체별로 몇 명의 학우들이 사용하는지 딕셔너리 타입으로 정리하고자 한다.
- ❖ 메일 서비스 업체 리스트를 인자로 받아 딕셔너리로 반환하는 함수 **tidy\_up\_company** 를 완성하시오. (반환 예시: {'Koogole': 2, 'Maver': 1} )
- ❖ **리스트 메서드 .count() 사용 시 감점**

```
email_list = [
    'Koogole', 'Koogole', 'JCloud', 'Koogole', 'GaKao',
    'School', 'Koogole', 'Maver', 'GaKao', 'Maver',
    'Koogole', 'GaKao', 'School', 'GaKao', 'JCloud',
    'School', 'GaKao', 'GaKao', 'Maver', 'Koogole'
]
```

## 문제 08 (problem08.py) – 5점

- ❖ 숫자 한 쌍이 리스트에 저장되어 있다고 한다.
- ❖ 어느 날 시스템의 오류로 짝이 맞지 않는 정수 하나가 버그로 해당 리스트에 들어가게 되었다.
- ❖ 이 때, 버그로 삽입된 짝이 맞지 않는 정수를 찾아 반환하는 **find\_solo** 함수를 완성하시오. (짝이 맞지 않는 정수는 1개만 존재한다.)
- ❖ **리스트 메서드 .count() 사용 시 감점**

## [파이썬 트랙] 1회차 월말평가 - Python



## 문제 09 (problem09.py) – 5점

- ❖ 이진수는 0,1로 모든 수를 표현하는 방식이다.
- ❖ 일상생활에서 사용하는 십진수 숫자를 이진수로 변경하기 위해서는 2로 나눈 몫을 2로 나누는 과정을 반복하며 나오는 나머지들을 사용한다.
- ❖ 십진수 숫자(n)를 인자로 받아 이진수로 변환하여 문자열로 반환하는 함수 `dec_to_bin` 을 완성하시오.
- ❖ 단, 재귀 함수를 이용하여 구현한다. (재귀 함수를 사용하지 않을 시 감점)

$$\begin{array}{r}
 2 \overline{) 55} \\
 2 \overline{) 27 \cdots 1} \\
 2 \overline{) 13 \cdots 1} \\
 2 \overline{) 6 \cdots 1} \\
 2 \overline{) 3 \cdots 0} \\
 \quad 1 \cdots 1
 \end{array}$$

$55 = 110111_2$

# [파이썬 트랙] 1회차 월말평가 - Python

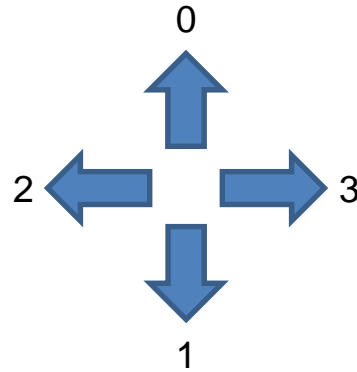


## 문제 10 (problem10.py) – 5점

- ❖ 게임 캐릭터가 움직일 수 있는 범위가 있으며, 이 제한된 구역을 넘어가지 않도록 검사하는 함수를 만들려고 한다.
- ❖ 캐릭터는 2차원 평면 ( $N * N$ )에서 이동한다. ( $0 < N \leq 100$ )
- ❖ 사용자의 키 입력에 따라 움직일 수 있으며, 입력은 0부터 3까지의 정수  $M$ 으로 주어진다. ( $0 \leq M \leq 3$ )
- ❖ 각 숫자별 이동 방향은 0:위, 1:아래, 2:왼쪽, 3:오른쪽을 의미한다.
- ❖ 캐릭터의 현재 위치(current) 는 튜플 (row, col) 형태로 주어지며, row와 col 은 각각 2차원 평면의 행과 열을 의미 한다. ( $0 \leq \text{row} < 100$ ,  $0 \leq \text{col} < 100$ )
- ❖ 현재 위치(current)에서 키 입력(M)으로 이동한 결과가 2차원 평면의 범위를 벗어난다면 False, 그렇지 않으면 True를 반환하는 함수 **is\_position\_safe** 를 완성하시오.


(0, 0)	(0, 1)	(0, 2)
(1, 0)	(1, 1)	(1, 2)
(2, 0)	(2, 1)	(2, 2)

좌표 예시




방향 예시

- 입력 예시 :  $N=3$ ,  $M=1$ ,  $\text{current} = (0, 0)$
- 결과 예시 : True





# [파이썬 트랙] 1회차 월말평가 - Python

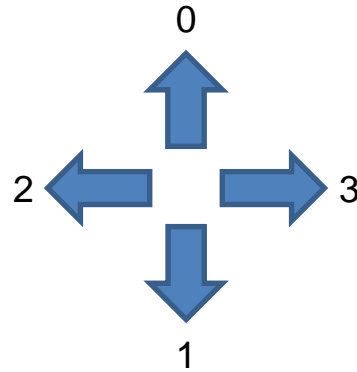


## 문제 11 (problem11.py) – 5점

- ❖ 게임 캐릭터가 움직일 수 있는 범위가 있으며, 이 제한된 구역을 넘어가지 않도록 검사하는 함수를 만들려고 한다.
- ❖ 캐릭터는 2차원 평면 ( $N * N$ )에서 이동한다. ( $0 < N \leq 100$ )
- ❖ 2차원 평면(matrix) 은 0과 1로 표현되어 있으며 **0은 빈 공간, 1은 현재 위치**를 나타낸다.
- ❖ 각 숫자별 이동 방향(M) 은 0:위, 1:아래, 2:왼쪽, 3:오른쪽을 의미한다. ( $0 \leq M \leq 3$ )
- ❖ 캐릭터의 이동은 이동 방향들의 리스트(move\_list)로 주어진다.
- ❖ 현재 위치를 찾은 후, 인자로 전달된 이동 리스트(move\_list) 에 따라 움직였을 때, 이동한 **최종 위치를 리스트 형태로 반환**하는 **get\_final\_position** 함수를 완성하시오.
- ❖ 최종 위치는 **[행, 열]** 형태로 반환하며, 이동 중 캐릭터가 **2차원 평면 밖을 넘어간다면 None**을 반환한다.

(0, 0)	(0, 1)	(0, 2)
(1, 0)	(1, 1)	(1, 2)
(2, 0)	(2, 1)	(2, 2)

좌표 예시



방향 예시

- 입력 예시 :  $N=3$ ,  $matrix=[[1, 0, 0], [0, 0, 0], [0, 0, 0]]$ ,  $move\_list=[1, 1, 3]$
- 결과 예시 :  $[2, 1]$

