

[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 성적 무효 및 중도 퇴소 처리)
- 2) 생성형 AI 사용 금지
- 3) VSCode 코파일럿 로그아웃 필수(코파일럿 로그인인 확인될 시 부정행위 처리)
- 4) 각 문제별로 스켈레톤 코드가 작성된 파일이 제공되며, 해당 코드 파일을 수정하여 정답 코드를 작성할 것 (단, 주어진 함수명은 수정불가)
- 5) 소스코드 유사도 판단 프로그램 기준 부정행위로 판단될 시, 0점 처리 및 학사 기준에 의거 조치 실시 예정
- 6) 테스트 케이스와는 별도로 채점 케이스가 존재함
- 7) 일부 문제에서 python 일부 내장 함수(예: min, max, len, sum) 사용 불가할 수 있음
(문제에 사용 불가 함수는 명시되어 있으며, 명시되지 않은 내장 함수는 자유롭게 사용 가능)
- 8) 사용자의 입력을 받는 input 함수는 절대 사용 금지
- 9) 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) – 16점

- ❖ 스포츠 비평가 스탯냥은 선수 기록을 보고 가장 높은 득점을 알고 싶다.
- ❖ 선수들의 득점이 담긴 양의 정수 리스트가 주어질 때, **최댓값(가장 높은 득점)을 정수로 반환하는 max_score 함수를 완성하시오.**
- ❖ **리스트에는 최소 1개 이상의 값이 있다.**
- ❖ 예시_01) [12, 7, 25, 18] → 25
- ❖ 예시_02) [12, 12, 12, 12] → 12
- ❖ 예시_03) [5] → 5
- ❖ **Python 내장 함수 max, sorted, 리스트 sort() 메서드 사용 시 감점**

문제 02 (problem02.py) – 16점

- ❖ 주어진 리스트는 여러 개의 문자열로 구성되어 있다.
- ❖ 이 **리스트에서 길이가 가장 긴 문자열을 반환하는 longest_string 함수를 완성하시오.**
- ❖ 만약 길이가 같은 문자열이 여러 개 있다면, 리스트에서 먼저 찾은 문자열을 반환한다.
- ❖ **문자열은 길이가 0인 것도 리스트에 존재할 수 있다.**
- ❖ print(longest_string(['apple', 'banana', 'cherry', 'date'])) → 'banana'
- ❖ print(longest_string(['cat', 'caterpillar', 'dog', 'elephant'])) → 'caterpillar'
- ❖ print(longest_string(['a', 'ab', 'abc', 'abcd'])) → 'abcd'
- ❖ **Python 내장 함수 len 사용시 감점**

[파이썬 트랙] 1회차 월말평가 - Python



문제 03 (problem03.py) – 16점

- ❖ OO 공방에서는 하루에 생산한 부품들의 불량률을 관리한다.
- ❖ 부품 검사 결과가 'pass' 또는 'fail' 문자열로 이루어진 리스트로 주어질 때, 전체 부품 대비 불량(fail) 부품 비율을 0~1 사이 float 로 반환하는 defect_rate 함수를 완성하시오.
- ❖ 검사 결과 리스트는 최소 1개 이상의 항목을 가진다.
- ❖ 'pass', 'fail' 이외 다른 문자열은 주어지지 않는다.
- ❖ 반환값에 대한 형식 지정(formatting)이나 반올림은 하지 않는다.
- ❖ 예시_01) ['pass', 'fail', 'pass', 'fail'] → 0.5
- ❖ 예시_02) ['pass', 'pass'] → 0.0
- ❖ 예시_03) ['fail', 'fail', 'fail'] → 1.0
- ❖ Python 내장 함수 sum, len, filter, 리스트 count 메서드 사용 시 감점

문제 04 (problem04.py) – 12점

- ❖ 시내 버스 차고지에서는 짝수 번호가 붙은 버스만 야간 점검 대상이다.
- ❖ 버스 번호가 담긴 정수 리스트가 주어질 때, 야간 점검 대상 버스(짝수 번호)의 수와 이들 번호의 총합을 튜플 (버스 개수, 번호_합) 형태로 반환하는 maintenance_stats 함수를 완성하시오.
- ❖ 버스 번호 리스트는 비어 있을 수 있다.
- ❖ 예시_01) [12, 7, 10, 5, 8] → (3, 30)
- ❖ 예시_02) [3, 5, 7] → (0, 0)
- ❖ 예시_03) [2, 4, 6, 8, 10, 12] → (6, 42)
- ❖ Python 내장 함수 sum, len, filter 사용 시 감점

[파이썬 트랙] 1회차 월말평가 - Python



문제 05 (problem05.py) – 12점

- ❖ 물품-가격 딕셔너리(price_map)와 고객 주문 목록 리스트(orders)가 주어질 때, 주문한 물품의 총 금액을 정수로 반환하는 calc_total 함수를 완성하시오.
- ❖ orders 에는 가격표에 없는 물품이 포함되지 않는다.
- ❖ 고객이 물품을 주문하지 않을 수도 있다.
- ❖ 예시_01) {'apple': 1000, 'pear': 800}, ['pear', 'apple', 'apple'] → 2800
- ❖ 예시_02) {'pen': 1200, 'note': 1500}, [] → 0
- ❖ 예시_03) {'apple': 1000, 'orange': 900, 'grape': 1200}, ['orange', 'orange'] → 1800
- ❖ Python 내장 함수 sum, map 사용 시 감점

문제 06 (problem06.py) – 12점

- ❖ 학생-점수 딕셔너리 scores 가 주어질 때, 평균 점수를 기준으로 학생을 A/B/C/D/F 다섯 등급으로 분류하고 {등급: [학생이름, ...]} 형태로 반환하는 grade_distribution 함수를 완성하시오.

구간	등급
90점 이상	A
80~89	B
70~79	C
60~69	D
59 이하	F

- ❖ 반환 딕셔너리에 학생이 없는 등급은 포함하지 않는다.
- ❖ 동일 등급 내 학생 순서는 입력 순서를 유지한다.
- ❖ 예시_01) {'Kim': 92, 'Lee': 75, 'Park': 88, 'Choi': 60}
 - {'A': ['Kim'], 'C': ['Lee'], 'B': ['Park'], 'D': ['Choi']}
- ❖ 예시_02) {'Min': 95, 'Oh': 93} → 'A': ['Min', 'Oh'] # 입력 순서 유지 & 등급이 하나만 존재

[파이썬 트랙] 1회차 월말평가 - Python



문제 07 (problem07.py) – 8점

- ❖ 주어진 문자열을 뒤집는 재귀 함수를 완성하시오.
- ❖ 함수 `reverse_string`은 문자열을 인자로 받아, 그 문자열을 뒤집은 결과를 반환한다.
- ❖ 반드시 재귀 함수 구조로 작성해야 한다.
- ❖ 반드시 재귀 함수를 이용해야 하며 재귀함수가 아니면 감점

문제 08 (problem08.py) – 8점

- ❖ 크기가 N인 2차원 정사각 행렬이 주어졌을 때, 가장 큰 값을 가지는 행, 열 좌표를 반환하는 `find_max_position` 함수를 완성하시오.
- ❖ 행렬의 크기 N은 2 이상 10 이하이다.
- ❖ 가장 큰 값이 여러 개 있을 경우 행의 인덱스가 가장 작은 좌표를 리스트로 반환한다.
- ❖ 가장 큰 값이 여러 개가 같은 행에 위치한 경우에는 열의 인덱스가 가장 작은 좌표를 리스트로 반환한다.