

Name:

Hemos ID:

Student ID (학번):

CSED-101 PROGRAMMING & PROBLEM SOLVING
Spring 2025 (Mock Test)
Midterm

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Total
Your Score													
Max Score	5	18	7	4	6	5	7	7	10	8	16	7	100

- Write down your name, hemos ID, and student ID.
- There are 14 pages in this midterm.
- Your answers must run correctly in **Python programming language** without error or warning. Otherwise, your answers will be considered incorrect. For example, it is ok to put more parentheses than needed in your answer, but it will be incorrect if you put fewer parentheses than needed.
- You must write your answer on the underline => _____. Scratches outside the underline will be ignored.
- The total score is 100
- This is a 3-hour exam.

명예서약서

나는 명예로운 포스테키안으로서 다음의 Honor Code 를 지킬 것을 맹세합니다.

정직과 타인에 대한 존중이 함께하는 포스테키안의 미래는 명예롭다.

2025 년 월 일

이름 (인)

1. (5 점) 아래 condition 의 결과가 true 이면 T, false 이면 F 로 답하시오.
단, 각 문항 별 1 점 배점, 오답일 경우 -1 점, 공백의 경우 0 점.

```
int a = 1, b = 2, c = 3, d = 4, e = 5, x = -9, y = -3, z = 0;
```

- 1.1. [_____] `a < c > b`
1.2. [_____] `4 * d / 5 % 3 && d`
1.3. [_____] `d == (e++) - 1`
1.4. [_____] `y == (a += b* = c -= 5)`
1.5. [_____] `z || x < (y - 5) && !y`

2. (18 점) 다음 문제를 읽고 답하시오.

- 2.1. (1 점) 다음 중 변수명으로 사용할 수 없는 것을 **모두** 고르시오.

① cs101 ② \$option ③ A4 ④ IF ⑤ _2dNum

- 2.2. (1 점) 다음 중 함수 선언(declaration)이 올바르게 된 것은?

- ① `def fun(x: int, y: int) -> int:`
② `def fun(x=3: int, y):`
③ `def fun(x: float, y=10: int) -> None:`
④ `def fun(x, y):`

- 2.3. (3 점) `x=1, y=2, z=3` 인 경우, 다음 코드를 실행 시킨 후의 `x, y, z` 의 값은?

<pre>if z > y: x -= 1 z -= 1 else: x -= 1 y -= 1</pre>	<p><code>x</code> : _____</p> <p><code>y</code> : _____</p> <p><code>z</code> : _____</p>
---------------------------------------------------------------------------	-------------------------------------------------------------------------------------------

2.4. (2 점) x=10, y=40 인 경우, 다음 코드를 실행 시킨 후의 x 의 값은?

<pre>while x < y: x += 9</pre>	x : _____
---------------------------------------	-----------

2.5. (3 점) x=0, y=1 인 경우, 다음 코드를 실행 시킨 후의 x, y 의 값은?

<pre>x = 10 while x > 0: y = y * x x -= 3</pre>	x : _____ y : _____
------------------------------------------------------------	------------------------

2.6. (2 점) x=0, y=0 인 경우, 다음 코드를 실행 시킨 후의 x, y 의 값은?

<pre>x = 0 while x < 10: if x % 2 == 0: x += 1 continue y += x x += 1</pre>	x : _____ y : _____
------------------------------------------------------------------------------------------------------------	------------------------

2.7. (4 점) x=0, y=0, z=0 인 경우, 다음 코드를 실행 시킨 후의 x, y, z 의 값은?

<pre>x = 0 while x < 10: y = x + 1 while y < 10: z += x + y y += 5 x += 3</pre>	x : _____ y : _____ z : _____
-------------------------------------------------------------------------------------------------------------------	-------------------------------------

2.8. (2 점) x=10, y=10, z=0 인 경우, 다음 코드를 실행 시킨 후의 z 의 값은?

<pre>while x > 0: z = x + y x -= 1</pre>	z : _____
-----------------------------------------------------	-----------

3. (7 점) 아래의 프로그램을 실행하였을 때, 화면에 출력되는 결과를 쓰시오.

```
ADD = 6 + 3

def f1(a, b):
    b = a + b
    a = 10
    return b

def f2(a, b, c):
    a -= 1
    b -= 1
    c -= 1
    return -a

def f3(a, b):
    for a in range(-4, 4):
        if a + 1:
            b += a
    return b

def swap(a, b):
    temp = a
    a = b
    b = temp
    print(f"{a} {b}")

def main():
    a, b, c = 2, 4, 5

    print(f"{a} {f1(a, b)}") # (1)

    c += f2(a, b, c)
    print(c) # (2)

    print(f3(c, 0)) # (3)

    swap(a, b) # (4)
    print(f"{a} {b}") # (5)

    print(ADD // 3) # (6)

if __name__ == "__main__":
    main()
```

출력 결과

(1) (1 점)	
(2) (1 점)	
(3) (2 점)	
(4) (1 점)	
(5) (1 점)	
(6) (1 점)	

4. (4 점) 아래 프로그램은 사용자로부터 두 숫자와 하나의 문자를 순서대로 입력 받아, 사칙 연산을 계산하는 프로그램이다. 이 프로그램이 정상적으로 작동할 수 있도록 빈 칸을 채우시오.

```
while True:
    a, b = map(float, input().split()) # 두 숫자 입력 받기
    c = input().strip() # 연산자 입력 받기

    if _____:
        print(f"{a} + {b} = {a + b}")
    elif _____:
        print(f"{a} - {b} = {a - b}")
    elif _____:
        print(f"{a} * {b} = {a * b}")
    elif _____:
        if b != 0:
            print(f"{a} / {b} = {a / b}")
        else:
            print("Cannot divide by zero")
    else:
        print("End")
        break
```

5. (6 점) 아래의 프로그램은 4 개의 정수를 입력 받아, 덧셈 및 뺄셈을 수행한 후 그 결과를 출력하는 프로그램이다. 아래의 실행 결과를 참조하여 빈 칸을 채우시오. 단, +, -, *, /, %와 같은 연산자를 사용하지 않고, 오직 **add** 와 **sub** 함수만을 이용해서 작성한다.

(실행예시)

```
Insert four integer values: 1 2 3 4
1+2+3+4=10
1-2-3-4=-8
```

```
def add(x, y):
    return x + y

def sub(x, y):
    return x - y

a, b, c, d = map(int, input("Insert four integer values: ").split())

e = _____

f = _____

print(f"{a}+{b}+{c}+{d}={e}")
print(f"{a}-{b}-{c}-{d}={f}")
```

6. (5 점) 아래 프로그램은 주사위 2 개를 굴려서 나온 숫자의 합이 짝수면 사용자가 이기고, 홀수면 지는 프로그램이다. 주사위에서 나올 수 있는 숫자의 범위는 1~6 이다. 프로그램이 정상 작동하도록 빈 칸을 채우시오.

```
import random

def randDice():
    return random.randint(1, 6)

def printResult(a):
    if ____:
        print(f"The sum of the 2 dice is {a}. You lose.")
    else:
        print(f"The sum of the 2 dice is {a}. You win.")

dice1 = randDice()
dice2 = randDice()

print(f"Dice 1: {dice1}, Dice 2: {dice2}")

_____
```

7. (7 점) 아래 프로그램은 3 자리 양의 정수를 랜덤하게 생성하려는 프로그램이다. 단, 각 자리의 숫자는 서로 중복되는 경우가 없다는 조건을 만족시켜야 한다. 빈 칸을 채우시오.

```
import random

TRUE = 1
FALSE = 0

def is_duplicate(n):
    n1 = _____
    n10 = _____
    n100 = _____

    check = FALSE

    if _____:
        check = TRUE

    return _____

num = None

random.seed() # 랜덤 시드를 설정

while True:
    num = _____

    if _____:
        break

print(f"{num}")
```


8. (7 점) 피보나치 수는 아래의 점화식으로 정의되는 수열이다.

$$F_n = \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F_{n-1} + F_{n-2} & \text{if } n > 1; \end{cases}$$

- 8.1. (4 점) 피보나치 수를 **int fibo (int n)** 이라는 함수를 사용하여 구현하고자 한다. 이 함수를 **recursion** 을 이용하여 작성하시오.

```
def fibo(n):
    if _____:
        return _____
    else:
        return _____

series_size = 12

print(f"First {series_size} Fibonacci numbers:")

for loop in range(series_size):
    if loop % 4:
        print(f", {fibo(loop):4d}", end="")
    else:
        print(f"\n{fibo(loop):4d}", end="")
```

- 8.2. (3 점) 위의 작성된 프로그램을 실행하였을 때, 화면에 출력되는 결과를 쓰시오.

9. (10 점) 다음 문제를 읽고 답하시오.

- 9.1. (5 점) 아래 프로그램은 사용자로부터 양의 정수 1 개를 입력 받아, 숫자의 순서를 거꾸로 출력하는 프로그램이다. 프로그램이 정상 작동하도록 빈 칸을 채우시오.

```
def reverseInt(n):
    if _____:
        print(n, end="")
    else:
        reverseInt(n % 10)
        _____

n = int(input())
reverseInt(n)
```

[실행 결과]

입력: 54321

출력: 12345

9.2. (5 점) 위의 함수 **reverseInt** 를 while 문을 이용하여 완성하시오. (재귀 함수로 작성할 수 없음)

```
def reverseInt(n):  
    while _____:  
        _____  
        _____  
  
n = int(input())  
reverseInt(n)
```

10. (8 점) 다음은 주어진 배열의 순서를 뒤집는 프로그램이다. 배열의 각 원소는 처음에는 배열에서의 자신의 위치 값을 할당한다. 프로그램이 정상적으로 작동하도록 빈칸을 채우시오. 추가적인 변수는 선언할 수 없으며 ARY_SIZE 를 최소 6 번 사용해야 한다.

```
ARY_SIZE = 10

def main():
    _____
    temp = 0

    # 초기 값을 지정해준다.
    for i in range(_____):
        _____
        print(_____, end=" ")

    print()

    # 배열의 순서를 뒤집어 저장한다.
    for i in range(_____):
        _____
        _____
        _____

    # 바뀐 값을 출력한다.
    for i in range(_____):
        print(_____, end=" ")

    print()

if __name__ == "__main__":
    main()
```

실행결과:

```
0 1 2 3 4 5 6 7 8 9
9 8 7 6 5 4 3 2 1 0
```

11. (16 점) 아래의 프로그램은 파일(input.txt)로부터 학생들의 시험 성적을 읽고 각 학생의 총점(평균으로 계산)과 학점 및 과목평균을 계산하여 화면에 출력하는 프로그램이다. 아래 빈칸을 채워 프로그램을 완성하시오. 단, 2 개의 변수만 선언해서 사용할 수 있다.

파일에는 줄(line) 단위로 한 학생의 학번, 중간고사 점수, 기말고사 점수가 순서대로 기록되어 있다.

(input.txt)

```
201301 89 78
201302 76 84
201303 67 42
201304 66 88
201305 98 99
```

(출력화면)

```
=====
학번 총점(학점)
=====
201301 83.5 (B)
201302 80.0 (B)
201303 54.5 (F)
201304 77.0 (C)
201305 98.5 (A)
=====
과목평균: 78.70
=====
```

```
def calc_average(x, y):
    return _____

def calc_grade(score):
    if score >= 90:
        return 'A'
    elif score >= 80:
        return 'B'
    elif score >= 70:
        return 'C'
    elif score >= 60:
        return 'D'
    else:
        return 'F'

def main():
    tot_ave = 0
    num_of_stu = 0

    try:
        with open(_____, 'r') as infile:
            print("=====")
            print("학번\t 총점(학점)")
            print("=====")

            while True:
                line = infile.readline()
                if not line:
                    break
                stu_id, mid, final = map(int, line.split())
```

```
        average = calc_average(mid, final)
        grade = calc_grade(average)

        print(f"{stu_id}\t{average:.1f} ({grade})")
        tot_ave += average
        num_of_stu += 1

    print("=====")
    print(f"과목평균: {tot_ave / num_of_stu:.2f}")
    print("=====")

except FileNotFoundError:
    print("cannot open input file.")
    return 1

if __name__ == "__main__":
    main()
```

12. (7 점) 아래의 프로그램은 배열의 원소 중 가장 큰 값을 찾아 그 값을 출력하는 프로그램이다.
빈 칸을 채우시오.

배열 list 는 아래의 그림과 같이 정수형의 원소로 구성되어 있다.

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
list	0	5	7	3	2	6	9	1	4	8

```
ARY_SIZE = 10

def get_max(arr, size):
    max_value = _____

    for i in range(1, size):
        if _____:
            _____

    return max_value

def main():
    # (2 점) 배열 list 를 위의 그림과 같이 초기화하시오.
    list = _____

    max_value = get_max(list, ARY_SIZE) # (2 점) 함수 get_max() 호출

    print(f"MAX = {max_value}")

if __name__ == "__main__":
    main()
```

실행 결과:

MAX = 9