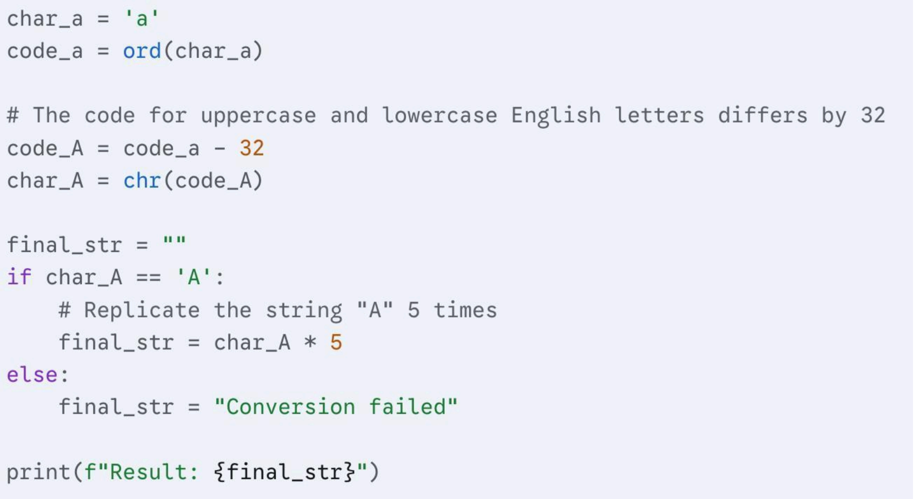
Homework 1.

- The file name of your homework (**in PDF**) should be in the format: “學號-python-作業編號.pdf”. For example: **00757999-python-hw1.pdf**
- Please submit your homework to Tronclass **before 23:59, October 19, 2025**

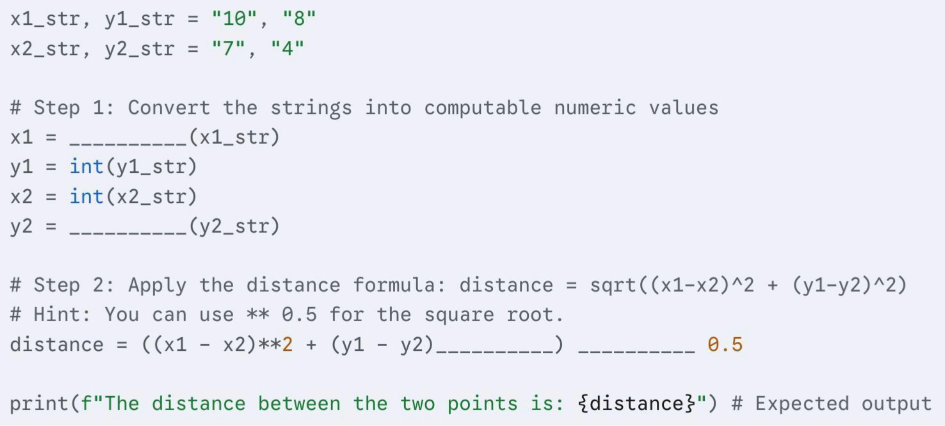
1. **(10%)** According to the rules of Python 3's round() function, what will be the value of x + y after executing x = round(4.5) and y = round(5.5)?
   (A) 9 (B) 10 (C) 11 (D) 10.0
   4+6=10
   Answer:(B)
2. **(10%)** Based on your knowledge of Python's built-in functions ord() (returns the code for a character) and chr() (returns the character for a code), what is the final output of the following code?



```
char_a = 'a'
code_a = ord(char_a)

# The code for uppercase and lowercase English letters differs by 32
code_A = code_a - 32
char_A = chr(code_A)

final_str = ""
if char_A == 'A':
    # Replicate the string "A" 5 times
    final_str = char_A * 5
else:
    final_str = "Conversion failed"

print(f"Result: {final_str}")
```

Answer:AAAAA

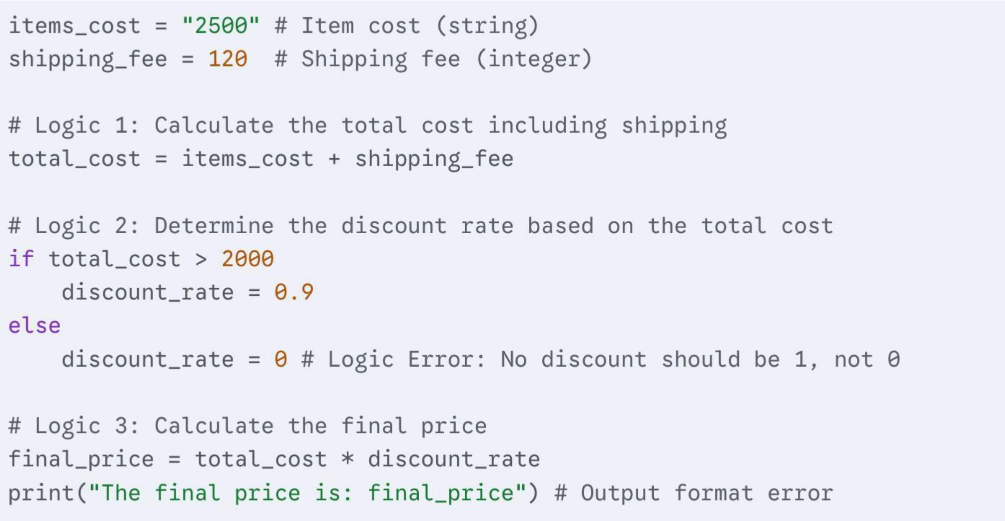
3. **(10%)** The code snippet below is intended to calculate the distance between two points on a plane. Please fill in the blanks with the most appropriate code to make it function correctly.



```
x1_str, y1_str = "10", "8"
x2_str, y2_str = "7", "4"

# Step 1: Convert the strings into computable numeric values
x1 = __________(x1_str)
y1 = int(y1_str)
x2 = int(x2_str)
y2 = __________(y2_str)

# Step 2: Apply the distance formula: distance = sqrt((x1-x2)^2 + (y1-y2)^2)
# Hint: You can use ** 0.5 for the square root.
distance = ((x1 - x2)**2 + (y1 - y2)__________) __________ 0.5

print(f"The distance between the two points is: {distance}") # Expected output
```

```
x1_str, y1_str = "10", "8"
x2_str, y2_str = "7", "4"
 # Step 1: Convert the strings into computable numeric values
 x1 = int(x1_str)
 y1 = int(y1_str)
 x2 = int(x2_str)
 y2 = int(y2_str)
# Step 2: Apply the distance formula
distance = ((x1 - x2)**2 + (y1 - y2)**2)**0.5
print(f"The distance between the two points is: {distance}")
```

4. **(15%)** The code below is intended to calculate a total purchase amount. The logic is as follows: if the items_cost is over 2000, a 10% discount is applied to the entire order (including shipping); otherwise, there is no discount. This code contains **type, syntax, and logical errors**. Please find all errors and provide the complete, corrected code.



```
items_cost = "2500" # Item cost (string)
shipping_fee = 120  # Shipping fee (integer)

# Logic 1: Calculate the total cost including shipping
total_cost = items_cost + shipping_fee

# Logic 2: Determine the discount rate based on the total cost
if total_cost > 2000
    discount_rate = 0.9
else
    discount_rate = 0 # Logic Error: No discount should be 1, not 0

# Logic 3: Calculate the final price
final_price = total_cost * discount_rate
print("The final price is: final_price") # Output format error
```

錯誤:

items_cost 要先轉換型別

折扣條件應該作用於「總金額」, 折扣率錯

print未正確插入變數

```
items_cost = "2500"     # Item cost (string)
shipping_fee = 120      # Shipping fee (integer)
# Convert item cost to number
items_cost = float(items_cost)
# Logic 1: Calculate the total cost including shipping
total_cost = items_cost + shipping_fee
# Logic 2: Determine the discount rate based on the total cost
if total_cost > 2000:
        discount_rate = 0.9   # 10% discount
```

```
else:
    discount_rate = 1.0   # No discount
# Logic 3: Calculate the final price
final_price = total_cost * discount_rate
# Correct output format
print(f"The final price is: {final_price:.2f}")
```

5. **(15%)** Write a Zodiac sign determination program. The program should prompt the user to enter a month and a day (both as integers). Based on the input, the program must determine and print the corresponding Zodiac sign. You must use an if-elif- else structure. (To simplify, you do not need to handle invalid input or non-existent dates).

Zodiac Date Reference:

- **Aquarius:** 1/20 - 2/18
- **Pisces:** 2/19 - 3/20
- **Aries:** 3/21 - 4/19
- **Taurus:** 4/20 - 5/20
- **Gemini:** 5/21 - 6/21
- **Cancer:** 6/22 - 7/22
- **Leo:** 7/23 - 8/22
- **Virgo:** 8/23 - 9/22
- **Libra:** 9/23 - 10/23
- **Scorpio:** 10/24 - 11/22
- **Sagittarius:** 11/23 - 12/21
- **Capricorn:** 12/22 - 1/19

Example Input/Output:

- Input Month: `8`, Input Day: `23` -> Output: `Your Zodiac sign is: Virgo`

```
month = int(input("Enter month (1-12): "))
day = int(input("Enter day: "))
if (month == 1 and day >= 20) or (month == 2 and day <= 18):
    sign = "Aquarius"
elif (month == 2 and day >= 19) or (month == 3 and day <= 20):
    sign = "Pisces"
elif (month == 3 and day >= 21) or (month == 4 and day <= 19):
    sign = "Aries"
elif (month == 4 and day >= 20) or (month == 5 and day <= 20):
    sign = "Taurus"
elif (month == 5 and day >= 21) or (month == 6 and day <= 21):
    sign = "Gemini"
elif (month == 6 and day >= 22) or (month == 7 and day <= 22):
    sign = "Cancer"
elif (month == 7 and day >= 23) or (month == 8 and day <= 22):
    sign = "Leo"
elif (month == 8 and day >= 23) or (month == 9 and day <= 22):
    sign = "Virgo"
elif (month == 9 and day >= 23) or (month == 10 and day <= 23):
    sign = "Libra"
```

```
elif (month == 10 and day >= 24) or (month == 11 and day <= 22):
    sign = "Scorpio"
elif (month == 11 and day >= 23) or (month == 12 and day <= 21):
    sign = "Sagittarius"
else:
    sign = "Capricorn"

print(f"Your Zodiac sign is: {sign}")
```

6. **(15%)** Write a program to parse employee data and calculate their **gross pay**.

- **Read Input:** The program should prompt the user for a single string of comma-separated employee data in the format `Name,TotalHours,HourlyRate`.
- **Parse Data:** Correctly separate the name (`str`), total hours (`int`), and hourly rate (`int`) from the input string.
- **Calculate Pay:**
  - Hours up to and including 40 are paid at the regular hourly rate.
  - Any hours over 40 are considered overtime and are paid at **1.5 times** the regular hourly rate.
- **Output:** Print the employee's name and their calculated gross pay, formatted to 2 decimal places.

Example Input/Output:

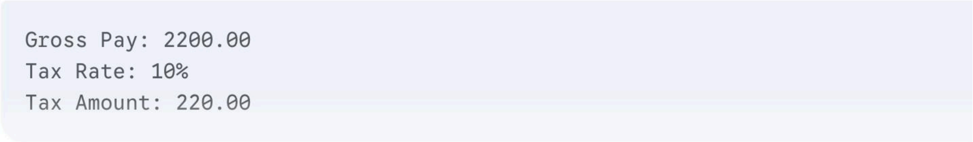
- **Input:** `David,50,40`
- **Output:** `Employee David's gross pay is: 2200.00`

```
data = input("Enter employee data (Name,TotalHours,HourlyRate): ")

name, total_hours, hourly_rate = data.split(",")

total_hours = int(total_hours)

hourly_rate = float(hourly_rate)

if total_hours <= 40:

    gross_pay = total_hours * hourly_rate

else:

    gross_pay = 40 * hourly_rate + (total_hours - 40) * hourly_rate * 1.5

print(f"Employee {name}'s gross pay is: {gross_pay:.2f}")
```

7. **(15%)** Write a program to calculate the **tax amount** based on **gross pay**.
   - **Read Input:** Prompt the user to enter a single number representing the gross pay.
   - **Tax Logic:**
     - Gross pay >= 3000: Tax rate is 20%
     - 1500 <= Gross pay < 3000: Tax rate is 10%
     - Gross pay < 1500: Tax rate is 5%
   - **Output:** Print the gross pay, the corresponding tax rate, and the calculated tax amount. Format monetary values to 2 decimal places and the rate as a percentage.

Example Input/Output:

- **Input:** `2200.00`
- **Output:**



```
Gross Pay: 2200.00
Tax Rate: 10%
Tax Amount: 220.00
```

```
gross_pay = float(input("Enter gross pay: "))
if gross_pay >= 3000:
    tax_rate = 0.20
elif gross_pay >= 1500:
    tax_rate = 0.10
else:
    tax_rate = 0.05
tax_amount = gross_pay * tax_rate
print(f"Gross Pay: {gross_pay:.2f}")
print(f"Tax Rate: {tax_rate * 100:.0f}%")
print(f"Tax Amount: {tax_amount:.2f}")
```

8. **(10%)** Write a program to generate a payslip with an exact format. Assume you are given the following three variables:

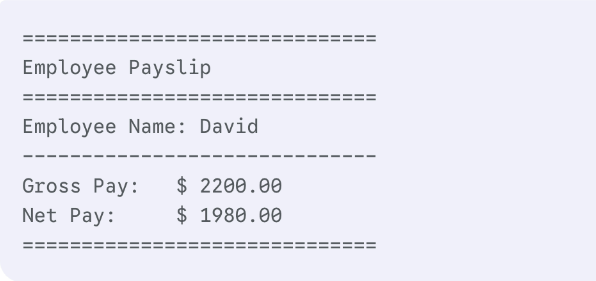
`name = "David"`

`gross_pay = 2200.00`

`net_pay = 1980.00`

Your task is to use only these three variables, along with the **alignment and width setting** features of f-strings or .format(), to print a layout that is **identical** to the example below.

Required Output Format:



```
==============================
Employee Payslip
==============================
Employee Name: David
------------------------------
Gross Pay:   $ 2200.00
Net Pay:     $ 1980.00
==============================
name = "David"
gross_pay = 2200.00
net_pay = 1980.00
print("=" * 27)
print("Employee Payslip")
print("=" * 27)
print(f"Employee Name: {name}")
print("-" * 27)
print(f"Gross Pay: {'$':>3} {gross_pay:>8.2f}")
print(f"Net Pay:   {'$':>3} {net_pay:>8.2f}")
print("=" * 27)
```