

Q1:

Question 1: Hourglass Pattern. Given a positive integer n , create an hourglass pattern with n rows at its longest width. The pattern should start with 1 asterisk on the first row, incrementing by 1 in each subsequent row until it reaches n . From there, it should decrement by 1 in each subsequent row until it reaches 1 asterisk again.

Output example:

```
*
**
***
****
*****
****
***
**
*
```

```
# Get the number of rows from user input
n=int(input("Enter the number of rows: "))

# Initialize row number to 1
row=1

# Outer while loop for iterating through rows
while row<=2*n-1:

    # Initialize counter variable for inner loop
    i=1

    # Inner while loop for printing asterisks on each row
    while i<=(n-abs(n-row)):

        # Print asterisk without newline
        print("*",end='')

        # Increment the counter variable
        i+=1

    # Print newline after each row of asterisks
    print()

    # Update variables for next row and move to the next row
    row+=1
```

The screenshot shows a VS Code editor with a Python file named `HW3_Q1.py`. The code is as follows:

```
1 # Get the number of rows from user input
2 n=int(input("Enter the number of rows: "))
3 # Initialize row number to 1
4 row=1
5 # Outer while loop for iterating through rows
6 while row<=n:
7     # Initialize counter variable for inner loop
8     i=1
9     # Inner while loop for printing asterisks on each row
10    while i<=(n-abs(n-row)):
11        # Print asterisk without newline
12        print("*",end=" ")
13        # Increment the counter variable
14        i=i+1
15    # Print newline after each row of asterisks
16    print()
17    # Update variables for next row and move to the next row
18    row=row+1
19
20
```

The terminal output shows the execution of the script:

```
Windows PowerShell
版权所有 (c) Microsoft Corporation. 保留所有权利。

安装最新的 PowerShell，了解新功能和改进！ https://aka.ms/PSWindows

PS C:\Users\WONG. Chi Kit\Desktop\code\python> & 'C:\Software\anaconda3\python.exe' 'c:\Users\WONG. Chi Kit\.vscode\extensions\ms-python.python-2023.16.0\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '51887' '->' 'c:\Users\WONG. Chi Kit\Desktop\code\python\Introduction_to_Computer_Science\Assignment\submit\UFUG1601_HW3_ChikitWONG\HW3_Q1.py\HW3_Q1.py'
Enter the number of rows: 5
*
* *
* * *
* * * *
* * * * *
```

Q2:

Question 2: Pyramid Pattern. Given a positive integer n , create a pyramid pattern with n rows. The pattern should start with 1 asterisk on the first row, centered in the output. Each subsequent row should increment the number of asterisks by 2, while maintaining the centered alignment, until it reaches the maximum number of asterisks at the n -th row.

Output example:

```
*
* *
* * *
* * * *
* * * * *
```

```

# Get the number of rows from user input
n=int(input("Enter the number of rows: "))

# Initialize the row number to 1
row=1

# Initialize the number of asterisks to print on the first row
num_asterisks=1

# Outer while loop for iterating through rows
while row<=n:

    # Calculate the number of spaces before asterisks
    space_count=n-row

    # Initialize the counter for printed asterisks on the current row
    asterisks_count=1

    # Inner while loop for printing spaces and asterisks on each row
    while space_count>0 or asterisks_count<=num_asterisks:

        # If there are spaces to print, print a space
        if space_count>0:
            print(" ",end='')

            # Decrease the space_count
            space_count-=1

        # If there are no more spaces to print, print an asterisk
        else:
            print("*",end="")

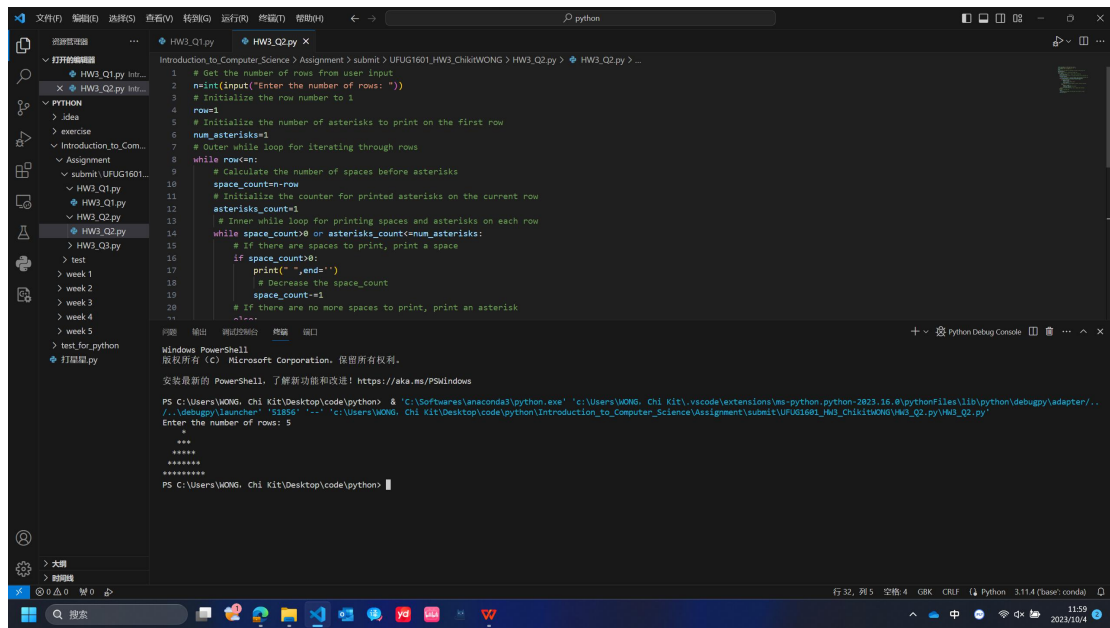
            # Increase the asterisks_count
            asterisks_count+=1

    # Print a newline after each row of spaces and asterisks
    print()

    # Move to the next row
    row+=1

    # Increment the number of asterisks for the next row by 2
    num_asterisks+=2

```



Q3:

Question 3: Diamond Pattern. Given an odd positive integer n , create a diamond pattern with n rows. The pattern should start with 1 asterisk on the first row, centered in the output. Each subsequent row should increment the number of asterisks by 2, while maintaining the centered alignment, until it reaches the maximum number of asterisks at the middle row ($\text{ceil}(n/2)$). From there, it should decrement the number of asterisks by 2 in each subsequent row, while maintaining the centered alignment, until it reaches 1 asterisk again on the last row.

Output example:

```
*
 *
***
****
 *
 *
```

```
import math
# Get the number of rows from user input
n=int(input("Enter the number of rows: "))
# Initialize variables
row=1
num_asterisks=1
# Calculate the middle row
middle_row=math.ceil(n/2)
# Outer loop for rows
while row<=n:
    # Calculate spaces before asterisks
    spaces=(n-num_asterisks)//2
    # Inner loop for printing spaces and asterisks
    for i in range(n):
        if i<spaces:
            print(" ", end="")
        elif i<spaces+num_asterisks:
            print("*", end="")
    # Print newline for the next row
    print()
    # Update variables for the next row
    if row<middle_row:
        num_asterisks+=2
    else:
        num_asterisks-=2
    row+=1
```

