

```
In [ ]: def factorial(n):
        if n == 0 or n == 1:
            return 1
        else:
            return n * factorial(n - 1)
        num = int(input("Enter a number: "))
        if num < 0:
            print("Factorial is not defined for negative numbers.")
        else:
            result = factorial(num)
            print(f"The factorial of {num} is: {result}")
```

```
In [*]: def is_prime(number):
        if number < 2:
            return False
        for i in range(2, int(number**0.5) + 1):
            if number % i == 0:
                return False
        return True
        num = int(input("Enter a number: "))
        if num < 0:
            print("Factorial is not defined for negative numbers.")
        else:
            result = factorial(num)
            print(f"The factorial of {num} is: {result}")
```

Enter a number:

```
In [*]: def is_palindrome(s):
        s = s.replace(" ", "").lower()
        return s == s[::-1]

        input_string = input("Enter a string: ")

        if is_palindrome(input_string):
            print(f"{input_string} is a palindrome.")
        else:
            print(f"{input_string} is not a palindrome.")
```

```
In [*]: def calculate_hypotenuse(side1, side2):

    hypotenuse = (side1**2 + side2**2)**0.5
    return hypotenuse

side1 = float(input("Enter the length of the first side: "))
side2 = float(input("Enter the length of the second side: "))

if side1 > 0 and side2 > 0:
    result = calculate_hypotenuse(side1, side2)
    print(f"The length of the hypotenuse is: {result:.2f}")
else:
    print("Please enter positive values for the sides of the triangle.")
```

```
In [*]: def calculate_char_frequency(input_string):
    char_frequency = {}
    for char in input_string:
        char_frequency[char] = char_frequency.get(char, 0) + 1

    return char_frequency
input_string = input("Enter a string: ")
frequency_dict = calculate_char_frequency(input_string)

print("Character frequencies:")
for char, count in frequency_dict.items():
    print(f"'{char}': {count}")
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: