NEURAL NETWORK DEEP LEARNING

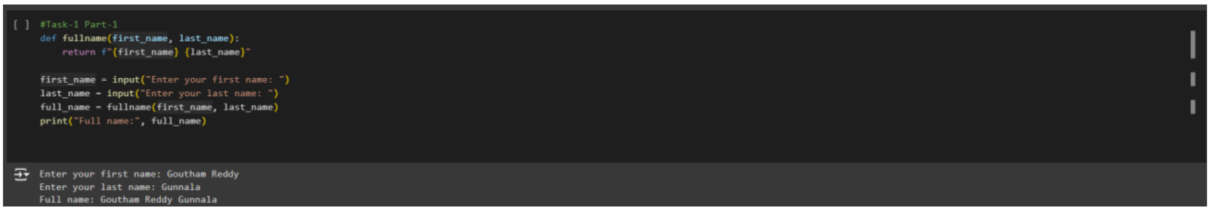
Name: Goutham Reddy Gunnala **Student ID:**700754593

Github link: https://github.com/Goutham-1012/ICP-2

Video link: https://drive.google.com/file/d/1tMqq1tbC7N6jzGOJY3-7oSENCbkji028/view?usp=sharing

Program Screenshots:

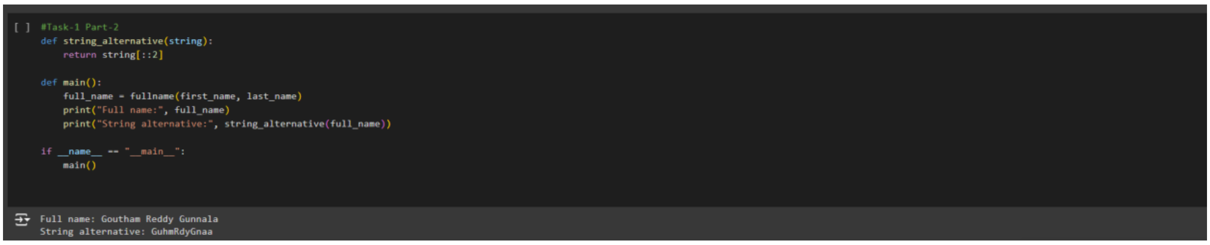
Program1:



```
#Task-1 Part-1
def fullname(first_name, last_name):
    return f"{first_name} {last_name}"

first_name = input("Enter your first name: ")
last_name = input("Enter your last name: ")
full_name = fullname(first_name, last_name)
print("Full name:", full_name)
```

```
Enter your first name: Goutham Reddy
Enter your last name: Gunnala
Full name: Goutham Reddy Gunnala
```



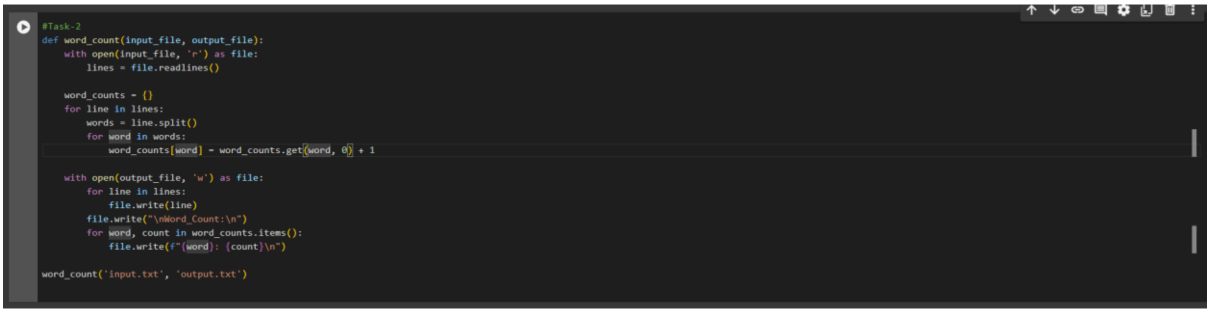
```
#Task-1 Part-2
def string_alternative(string):
    return string[::2]

def main():
    full_name = fullname(first_name, last_name)
    print("Full name:", full_name)
    print("String alternative:", string_alternative(full_name))

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

```
Full name: Goutham Reddy Gunnala
String alternative: GuhmRdyGnaa
```

Program2:



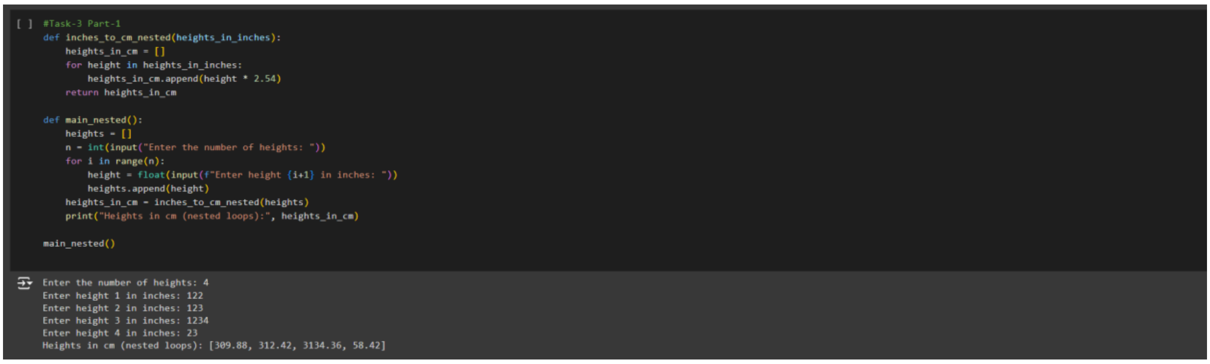
```
#Task-2
def word_count(input_file, output_file):
    with open(input_file, 'r') as file:
        lines = file.readlines()

    word_counts = {}
    for line in lines:
        words = line.split()
        for word in words:
            word_counts[word] = word_counts.get(word, 0) + 1

    with open(output_file, 'w') as file:
        for line in lines:
            file.write(line)
        file.write("\nWord_Count:\n")
        for word, count in word_counts.items():
            file.write(f"{word}: {count}\n")

word_count('input.txt', 'output.txt')
```

Program 3:

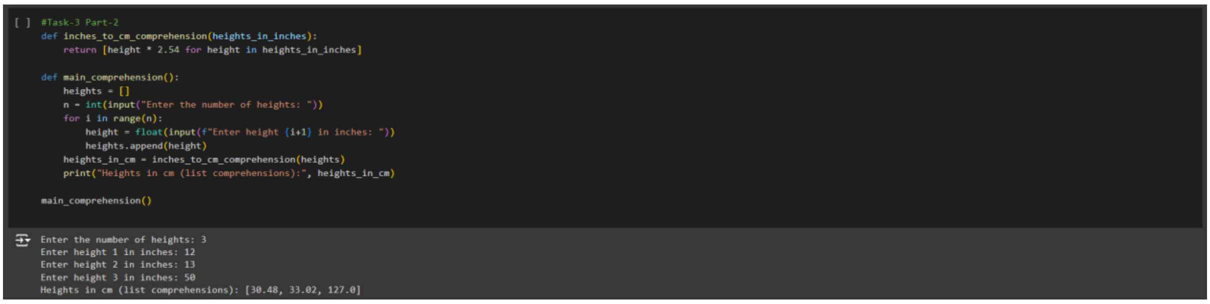


```python
#Task-3 Part-1
def inches_to_cm_nested(heights_in_inches):
    heights_in_cm = []
    for height in heights_in_inches:
        heights_in_cm.append(height * 2.54)
    return heights_in_cm

def main_nested():
    heights = []
    n = int(input("Enter the number of heights: "))
    for i in range(n):
        height = float(input(f"Enter height {i+1} in inches: "))
        heights.append(height)
    heights_in_cm = inches_to_cm_nested(heights)
    print("Heights in cm (nested loops):", heights_in_cm)

main_nested()
```

```
Enter the number of heights: 4
Enter height 1 in inches: 122
Enter height 2 in inches: 123
Enter height 3 in inches: 1234
Enter height 4 in inches: 23
Heights in cm (nested loops): [309.88, 312.42, 3134.36, 58.42]
```



```python
#Task-3 Part-2
def inches_to_cm_comprehension(heights_in_inches):
    return [height * 2.54 for height in heights_in_inches]

def main_comprehension():
    heights = []
    n = int(input("Enter the number of heights: "))
    for i in range(n):
        height = float(input(f"Enter height {i+1} in inches: "))
        heights.append(height)
    heights_in_cm = inches_to_cm_comprehension(heights)
    print("Heights in cm (list comprehensions):", heights_in_cm)

main_comprehension()
```

```
Enter the number of heights: 3
Enter height 1 in inches: 12
Enter height 2 in inches: 13
Enter height 3 in inches: 50
Heights in cm (list comprehensions): [30.48, 33.02, 127.0]
```