Part 6 – String & List

1. Problem 1 – Compare String

```
def compare(a, b):
    result = ""
    for char1, char2 in zip(a, b):
        if char1 == char2:
            result += char1
        else:
            break
    return result

print(compare("AKA", "AKASHI"))
print(compare("KANGOORO", "KANG"))
print(compare("KI", "KIJANG"))
print(compare("KUPU-KUPU", "KUPU"))
print(compare("ILALANG", "ILA"))
```

2. Problem 2 – Caeshar Cipher

```
def caesar(offset, input_str):
    result = ""

    for char in input_str:
        if char.islower():
            new_char = chr((ord(char) - ord('a') + offset) % 26 + ord('a'))
            result += new_char
        elif char.isupper():
            new_char = chr((ord(char) - ord('A') + offset) % 26 + ord('A'))
            result += new_char
        else:
            result += char

    return result
```

```
print(caesar(3, "abc"))
print(caesar(2, "alta"))
print(caesar(10, "alterraacademy"))
print(caesar(1, "abcdefghijklmnopqrstuvwxyz"))
print(caesar(1000, "abcdefghijklmnopqrstuvwxyz"))
```

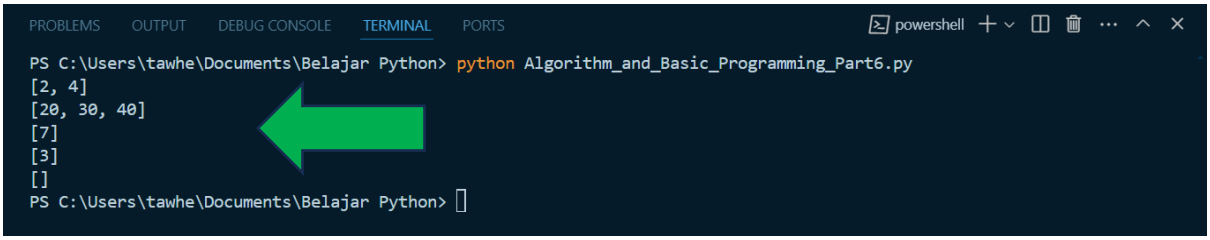
3. Problem 3 – Array Unique

```
def array_unique(arrayA, arrayB):
    setB = set(arrayB)

    unique_elements = [item for item in arrayA if item not in setB]

    return unique_elements

print(array_unique([1, 2, 3, 4], [1, 3, 5, 10, 16]))
print(array_unique([10, 20, 30, 40], [5, 10, 15, 59]))
print(array_unique([1, 3, 7], [1, 3, 5]))
print(array_unique([3, 8], [2, 8]))
print(array_unique([1, 2, 3], [3, 2, 1]))
```



4. Problem 4 – Maximum Sum Subarray of Size K

```
def find_max_sum_sub_array(k, arr):
    max_sum = 0
    window_sum = 0

    for i in range(k):
        window_sum += arr[i]

    max_sum = window_sum

    for i in range(k, len(arr)):
        window_sum += arr[i] - arr[i - k]
```

```
        max_sum = max(max_sum, window_sum)

    return max_sum

print(find_max_sum_sub_array(3, [2, 1, 5, 1, 3, 2]))
print(find_max_sum_sub_array(2, [2, 3, 4, 1, 5]))
print(find_max_sum_sub_array(2, [2, 1, 4, 1, 1]))
print(find_max_sum_sub_array(3, [2, 1, 4, 1, 1]))
print(find_max_sum_sub_array(4, [2, 1, 4, 1, 1]))
```

5. Problem 5 – Remove Duplicates

```
def remove_duplicates(arr):
    if not arr:
        return 0

    unique_index = 1

    for i in range(1, len(arr)):
        if arr[i] != arr[unique_index - 1]:
            arr[unique_index] = arr[i]
            unique_index += 1

    return unique_index

print(remove_duplicates([2, 3, 3, 3, 6, 9, 9]))
print(remove_duplicates([2, 3, 4, 5, 6, 9, 9]))
print(remove_duplicates([2, 2, 2, 11]))
print(remove_duplicates([2, 2, 2, 11]))
print(remove_duplicates([1, 2, 3, 11, 11]))
```