Week 3 Programming Assignment Submission

M. Zain ul Abideen

cosmicodepk@gmail.com

+923344770224

Submitted on: July 10, 2025

1 Task 1: Merge Sort for Integers

1.1 Code

```python
def merge_sort(arr):
    if len(arr) <= 1:
        return arr
    mid = len(arr) // 2
    left = merge_sort(arr[:mid])
    right = merge_sort(arr[mid:])
    return merge(left, right)

def merge(left, right):
    result = []
    i = j = 0
    while i < len(left) and j < len(right):
        if left[i] <= right[j]:
            result.append(left[i])
            i += 1
        else:
            result.append(right[j])
            j += 1
    result.extend(left[i:])
    result.extend(right[j:])
    return result

if __name__ == "__main__":
    numbers = [64, 34, 25, 12, 22, 11, 90]
    print("Original list:", numbers)
    sorted_list = merge_sort(numbers)
    print("Sorted list:", sorted_list)
```

Listing 1: Python code to sort a list of integers using merge sort

1.2 Output

```
Original list: [64, 34, 25, 12, 22, 11, 90]
Sorted list: [11, 12, 22, 25, 34, 64, 90]
```

2 Task 2: Reverse a List Without Built-in Functions

2.1 Code

```python
def reverse_list(arr):
    length = 0
    for _ in arr:
        length += 1
    reversed_arr = [0] * length
    for i in range(length):
        reversed_arr[i] = arr[length - 1 - i]
```

```
    return reversed_arr

if __name__ == "__main__":
    original_list = [1, 2, 3, 4, 5]
    print("Original list:", original_list)
    reversed_list = reverse_list(original_list)
    print("Reversed list:", reversed_list)
```

Listing 2: Python code to reverse a list without built-in functions

2.2 Output

```
Original list: [1, 2, 3, 4, 5]
Reversed list: [5, 4, 3, 2, 1]
```

3 Task 3: Find the Longest Word in a Sentence

3.1 Code

```
import string

def find_longest_word(sentence):
    translator = str.maketrans('', '', string.punctuation)
    clean_sentence = sentence.translate(translator)
    words = clean_sentence.split()
    if not words:
        return ""
    longest_word = words[0]
    for word in words:
        if len(word) > len(longest_word):
            longest_word = word
    return longest_word

if __name__ == "__main__":
    sentence = input("Enter a sentence: ")
    result = find_longest_word(sentence)
    if result:
        print("Longest word:", result)
    else:
        print("No words found in the sentence.")
```

Listing 3: Python code to find the longest word in a sentence

3.2 Output

```
Enter a sentence: Hello, world! This is a test sentence.
Longest word: sentence
```

4 Task 4: Check if a String is a Palindrome

4.1 Code

```python
def is_palindrome(s):
    cleaned = s.lower().replace(" ", "")
    left, right = 0, len(cleaned) - 1
    while left < right:
        if cleaned[left] != cleaned[right]:
            return False
        left += 1
        right -= 1
    return True

if __name__ == "__main__":
    user_input = input("Enter a string: ")
    if is_palindrome(user_input):
        print(f'"{user_input}" is a palindrome.')
    else:
        print(f'"{user_input}" is not a palindrome.')
```

Listing 4: Python code to check if a string is a palindrome

4.2 Output

```
Enter a string: A man a plan a canal Panama
"A man a plan a canal Panama" is a palindrome.
```

5 Task 5: Find the Most Frequent Word in a Text File

5.1 Code

```python
import string

def most_frequent_word(file_path):
    try:
        with open(file_path, 'r') as file:
            text = file.read()
        translator = str.maketrans('', '', string.punctuation)
        clean_text = text.translate(translator).lower()
        words = clean_text.split()
        if not words:
            return None, 0
        word_count = {}
        for word in words:
            if word:
                word_count[word] = word_count.get(word, 0) + 1
        max_word = max(word_count, key=word_count.get)
        max_count = word_count[max_word]
        return max_word, max_count
```

```
    except FileNotFoundError:
        print(f"Error: File '{file_path}' not found.")
        return None, 0
    except Exception as e:
        print(f"Error: {str(e)}")
        return None, 0

if __name__ == "__main__":
    file_path = input("Enter the path to the text file: ")
    word, count = most_frequent_word(file_path)
    if word:
        print(f"Most frequent word: '{word}' (appears {count}
           times)")
    else:
        print("No valid words found in the file.")
```

Listing 5: Python code to find the most frequent word in a text file

5.2 Output

```
Enter the path to the text file: sample.txt
Most frequent word: 'the' (appears 2 times)
```