Python Lists and Strings Coding Questions

Total Marks: 30
(10 questions × 3 marks each)

Passing Marks: 12
Excellent Marks: 24 and above

Difficulty Level: Easy
Time: 60 minutes

Instructions

- Questions are marked based on difficulty:
- - Easy questions: 3 marks each
- - Medium questions: 5 marks each
- - Difficult questions: 10 marks each
- Write clean, well-commented code
- Include basic error handling where appropriate
- Test your code with the given test cases
- Bonus question is for extra practice and does not count towards total marks

---

1: String Length Check [3 marks]

Write a function `check_string_length(s)` that returns True if the string length is between 5 and 10 characters (inclusive), False otherwise.

Example:

```
Input: "Hello"
Output: True

Input: "Hi"
Output: False

# Write your answer here
```

Answer:

```
def check_string_length(string):
   for char in string:
      If 10>len(string)>5:
       return True
   else:
      return False
print(check_string_length("Hello"))
# Write your answer here
```

2: List Sum [3 marks]

Write a function `sum_list(numbers)` that takes a list of numbers and returns their sum. Handle empty lists by returning 0.

Example:

```
Input: [1, 2, 3, 4, 5]
Output: 15

Input: []
Output: 0

# Write your answer here
```

Answer:

```
     def sum_list(numbers):
      total = 0
   for num in numbers:
      total += num
   return total
print(sum_list([]))
```

```
# Write your answer here
```

3: String Case Converter [3 marks]

Write a function `convert_case(s)` that converts a string to uppercase if it contains more uppercase letters, and lowercase if it contains more lowercase letters. If equal, return the original string.

Example:

```
Input: "HeLLo"
Output: "HELLO"

Input: "hello"
Output: "hello"
```

```
# Write your answer here
```

Answer:

```
def convert_case(string):
 for char in string:
    if max(string) == max(string).upper():
      return string.upper()
    if max(string) == max(string).lower():
      return string.lower()
    if max(string).upper() == max(string).lower():
      return string
print(convert_case("Hello"))
```

```
# Write your answer here
```

4: List Duplicate Remover [3 marks]

Write a function `remove_duplicates(lst)` that removes duplicate elements from a list while preserving the order of first occurrence.

Example:

```
Input: [1, 2, 2, 3, 3, 4, 5, 5]
Output: [1, 2, 3, 4, 5]

Input: ["apple", "banana", "apple", "cherry"]
Output: ["apple", "banana", "cherry"]

# Write your answer here
```

Answer:

```
def remove_duplicates(lst):
 original_set = []
 for char in lst:
     if char not in original_set:
       original_set += [char]
 return original_set
print(remove_duplicates([1,2,3,3,3,4,4,5,5,5]))
# Write your answer here
```

5: String Word Counter [3 marks]

Write a function `count_words(s)` that counts the number of words in a string. Words are separated by spaces.

Example:

```
Input: "Hello world"
Output: 2

Input: "This is a test sentence"
Output: 5
```

```
# Write your answer here
```

Answer:

```
def count_words(s):
 count = 0
 for char in s:
     count += 1
     return char
print(count_words("Hello World"))
# Write your answer here
```

6: List Even Numbers [3 marks]

Write a function `get_even_numbers(lst)` that returns a new list containing only the even numbers from the input list.

Example:

```
Input: [1, 2, 3, 4, 5, 6, 7, 8]
Output: [2, 4, 6, 8]

Input: [1, 3, 5, 7]
Output: []

# Write your answer here
```

Answer:

```
def get_even_numbers(lst):
  for numbers in lst:
    if numbers % 2 == 0:
      return numbers
print(get_even_numbers([1,2,3,4,5,6,7,8]))
# Write your answer here
```

7: String Reverser [3 marks]

Write a function `reverse_string(s)` that reverses a string without using the built-in reverse() method.

Example:

```
Input: "hello"
Output: "olleh"

Input: "python"
Output: "nohtyp"

# Write your answer here
```

Answer:

```
def reversed_string(string):


 val = 0

 for char in string:

   val = string[::-1]

   return val

print(reversed_string("Python"))

# Write your answer here
```

8: List Element Counter [3 marks]

Write a function `count_element(lst, element)` that counts how many times a specific element appears in a list.

Example:

```
Input: lst = [1, 2, 2, 3, 2, 4], element = 2
Output: 3
```

```
Input: lst = ["a", "b", "a", "c", "a"], element = "a"
Output: 3

# Write your answer here
```

Answer:

```
def count_element(lst,ele):
  count = 0
  for num in lst:
    count += num
    return count
print(count_element([1,2,2,3,3,3,4,5],ele=3))
# Write your answer here
```

9: String Vowel Counter [3 marks]

Write a function `count_vowels(s)` that counts the number of vowels (a, e, i, o, u) in a string, ignoring case.

Example:

```
Input: "Hello World"
Output: 3

Input: "Python Programming"
Output: 4

# Write your answer here
```

Answer:

```
def count_vowels(string):
  vowels = "aeiouAEIOU"
  count = 0
  for char in string:
     if char in vowels:
```

```
        count += 1
    return count
print(count_vowels("Hello World "))
# Write your answer here
```

Bonus Challenge [Extra Practice]

[Extra Practice]
Write a function `is_anagram(s1, s2)` that checks if two strings are anagrams of each other. An anagram is a word formed by rearranging the letters of another word, using all the original letters exactly once. For example, "listen" and "silent" are anagrams because they contain the same letters in different order.

Example:

```
Input: s1 = "listen", s2 = "silent"
Output: True

Input: s1 = "hello", s2 = "world"
Output: False

# Write your answer here
```

Answer:

```
def is_anagram(s1,s2):
    s1=s1.replace(" ","").lower()
    s2=s2.replace(" ","").lower()
    return sorted(s1) == sorted(s2)
print(is_anagram("listen","silent"))
# Write your answer here
```

Evaluation Criteria

- Correctness (50%)
- - Function works as expected (30%)
- - Handles edge cases correctly (20%)

- Code Readability (30%)
- - Clear variable names (10%)
- - Proper comments (10%)
- - Clean code structure (10%)
- Basic Error Handling (20%)
- - Input validation (10%)
- - Appropriate error messages (10%)

Marking Scheme

- Easy questions: 3 marks each
- Medium questions: 5 marks each
- Difficult questions: 10 marks each
- Total marks for the paper: 30
- Passing marks: 12
- Excellent marks: 24 and above

Good Luck!