**CSE 212 – Programming with Data Structures**

**W03 Prove – Response Document**

|  |  |
| --- | --- |
| **Name:** | Ben Cornia |
| **Date:** | 5/7/22 |
| **Teacher:** | Brother Alvey |

*It is a violation of BYU-Idaho Honor Code to post or share this document with others or to post it online. Storage into a personal and private repository (e.g. private GitHub repository, unshared Google Drive folder) is acceptable.*

**Question 1: From Part 1, describe what the Mystery Stack 1 code does and how the use of a stack helps in the implementation.**

The function takes a string as input and returns the string reversed. It loops through the string and appends each letter to the stack. Then while the stack is full it removes the last value from the stack and adds it to a string value. Once the stack is empty. The string value is returned.

The stack allows the user to only remove values from the end of the stack. Essentially, the string is removed from the stack in reverse making it simple to construct it backwards.

**Question 2: From Part 1, what are the three outputs from the Mystery Stack 1 code for the following three different inputs?**

* **racecar -> racecar**
* **stressed -> desserts**
* **a nut for a jar of tuna -> anut fo raj a rof tun a**

**Question 3: From Part 2, describe what the Mystery Stack 2 code does and how the use of a stack helps in the implementation.**

**The function takes a string of numbers and operators and returns the result. The stack allows you to maintain a specific order of execution. You could construct a string of values and operators such that the order of operations is honored (such as ensuring multiplication happens before subtraction).**

**Question 4: From Part 2, answer the following regarding what the Mystery Stack 2 code does:**

* **What will the result be if the input parameter is: 5 3 7 + \***

**50.0**

* **What will the result be if the input parameter is: 6 2 + 5 3 - /**

**4.0**

* **What input parameter would result in the display of “Invalid Case 1!”**

**5 + 2 ( less than two numbers before an operator)**

* **What input parameter would result in the display of “Invalid Case 2!”**

**5 0 / (dividing by 0)**

* **What input parameter would result in the display of “Invalid Case 3!”**

**5 k – (having a word character present in the string)**

* **What input parameter would result in the display of “Invalid Case 4!”**

an empty string or: “”

OR

Not enough operators: 5 3 6 -