Lab #09

***Alexander Fan***

***100229106***

CPSC 1150 - 003

Instructor: H. Darbandi

Lab Title: Strings and Arrays

Date Completed: Jul 02, 2020

Department: CSIS

Program: Strings and Arrays

File Name: Lab09.java

Purpose: Working with Strings and Arrays to turn numbers to words

Technical Information:

(You should fill the following information based on compiler and computer you are using).

Compiler: Java SDK version 14

Computer: AMD Ryzen 5 2600 3.40 GHz, 16 GB ram, 64-bit processor, Java SDK 14

Operating System: Windows 10

Language: Java

Program Logic (Pseudocode)

Algorithm:

**START**

0.1 static final String[] *unitsDigit* = {"zero", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine", ten", "eleven", "twelve", "thirteen", "fourteen", "fifteen", "sixteen", "seventeen", "eighteen", "nineteen"};

0.2 static final String[] *tensDigit* = {“”, “”, "twenty", "thirty", "forty", "fifty", "sixty", "seventy", "eighty", "ninety"};

0.3 static final String[] tensPower = {“”, “thousand”, “million”, “billion”, “trillion”, “quadrillion”}

*Public static String numToWordLessThanOneThousand(long number)* ***(phase 1)***

1. String result = “”;
2. If(number == 0)
   1. Return result = “zero”;
3. Int num = (int) number % 100;
4. If(num < 20)
   1. result = unitsDigit[num];
5. Else
   1. result += tensDigit[num / 10] + “ ” + unitsDigit[num % 10];
6. if(number > 100)
   1. result = unitsDigit[(int) number / 100] + “ hundred ” + result;
7. result = result.trim();
8. return result;

*Public static String convertNumToWord(long number)*

1. String result = “”;
2. Int tensPowerIndex = 0;
3. Do{
   1. Int num = (int) (number % 1000)
   2. If (tensPowerIndex > 0)
      1. Result = “, and ” + result;
   3. String str = numToWordLessThanOneThousand(num);
   4. Result = str + “ “ + placeValues[index] + result;
   5. tensPowerIndex+=1;
   6. number = number / 1000;
4. }while(number > 0);
5. Result = result.trim();
6. return result;

**END**

Generate your test cases based on the specifications in your lab assignment. Follow following format for each test case: (Refer to external document of your previous lab)

*purpose*

*input*

*output*

*expected value*

*passed or failed*

**Test Cases:**

***Test Case 1:***

Input: 0

Expected Output: zero

Actual Output: The translated number is zero

Passed

***Test Case 2:***

Input: 100

Expected Output: one hundred

Actual Output: The translated number is one hundred

Passed

***Test Case 3:***

Input: 1324651

Expected Output: one million, and three hundred twenty four thousand, and six hundred fifty one

Actual Output: The translated number is one million, and three hundred twenty four thousand, and six hundred fifty one.

Passed

***Test Case 4:***

Input: 56046541980616

Expected Output: fifty six trillion, and forty six billion and five hundred forty one million, and nine hundred eighty thousand, and six hundred sixteen

Actual Output: The translated number is fifty six trillion, and forty six billion, and five hundred forty one million, and nine hundred eighty thousand, and six hundred sixteen.

Passed